B
60

THESE

présentée à

L' UNIVERSITE PIERRE ET MARIE CURIE (PARIS VI)

pour l'obtention du

DOCTORAT D'ETAT ES SCIENCES

Spécialité : MATHEMATIQUES

Mention : INFORMATIQUE

par M. *Gérard NOGUEZ*

Sujet de la thèse : ETUDE D'UN MODELE TEMPOREL DES SYSTEMES SEQUENTIELS

Soutenue le Décembre 1975 devant la Commission

composée de : Président M. le Professeur J. ARSAC

Examinateurs : MM. les Professeurs

J. HEBENSTREIT

S. HYDER

L. NOLIN

J. C. SIMON

THESE

présentée à

L' UNIVERSITE PIERRE ET MARIE CURIE (PARIS VI)

pour l'obtention du

DOCTORAT D'ETAT ES SCIENCES

Spécialité : MATHEMATIQUES

Mention : INFORMATIQUE

par M. *Gérard NOGUEZ*

Sujet de la thèse : ETUDE D'UN MODELE TEMPOREL DES SYSTEMES SEQUENTIELS

Soutenue le Décembre 1975 devant la Commission

composée de : Président M. le Professeur J. ARSAC

Examinateurs : MM. les Professeurs

J. HEBENSTREIT

S. HYDER

L. NOLIN

J. C. SIMON

UNIV.
LEIDEN
BIBL.

A MARYSE

Digitized by the Internet Archive
in 2026

https://archive.org/details/IA41548119_0010

Qu'il me soit permis de remercier Monsieur le Professeur ARSAC qui m'a fait l'honneur de présider ce Jury et m'a prodigué ses encouragements, Monsieur HEBENSTREIT pour les discussions que nous avons eues et qui m'ont guidé dans la rédaction de cette thèse , Monsieur le Professeur HYDER qui tout au long de ses séjours à Paris a suivi et discuté le développement de ces travaux , Monsieur le Professeur NOLIN pour l'esprit de rigueur qu'il m'a enseigné tant pour la terminologie que pour la formalisation des problèmes logiques, Monsieur le Professeur SIMON pour l'intérêt qu'il a porté à ces travaux , Messieurs SUCHARD et ETIEMBLE dont les travaux sont à l'origine de cette étude.

Que tous ceux enfin qui ont permis l'achèvement de ce travail soient ici remerciés , particulièrement le secrétariat et l'imprimerie de l'Institut de Programmation.

INTRODUCTION

La conception d'un ordinateur est une tâche délicate si l'on considère qu'elle se trouve à l'intersection de deux domaines difficilement conciliables: le domaine de l'abstrait et celui du concret. Le principal danger qui guette "l'architecte de la machine" est de confondre dans son raisonnement les deux domaines.

La notion de temps nous paraît être le lieu privilégié de cette confusion: si nous pouvons nous servir par la pensée de tout phénomène comme point de repère - par exemple un signal qui véhiculerait des résultats - la notion de temps prend par contre dans la réalité l'aspect banal d'un signal qui n'a ni plus ni moins de propriétés que les autres signaux.

Afin de mettre en évidence ce type de difficulté, nous avons jugé bon d'expliquer nos motivations dans un premier chapitre. Nous proposons ensuite dans un second chapitre un modèle temporel des systèmes séquentiels. Les autres chapitres sont consacrés à l'application de ce modèle aux principaux problèmes temporels que l'on rencontre lors de la conception d'un ordinateur. Certaines applications sont renvoyées en annexe afin de ne pas trop alourdir l'exposé.

ETUDE D'UN MODELE TEMPOREL
DES SYSTEMES SEQUENTIELS

I - Motivations
II - Le modèle proposé
III - Application aux systèmes à mémoire
IV - Application aux systèmes à temps propre

Annexe - Application aux systèmes combinatoires

CHAPITRE I

Dans ce chapitre, nous essayons d'expliquer et de justifier le choix des fondements de notre modèle temporel. Nous rappelons d'abord la définition usuelle la plus générale des systèmes séquentiels. Nous analysons ensuite les problèmes temporels inhérents à cette définition. Nous étudions alors les solutions apportées par les différentes théories des machines séquentielles. Cela nous amène à la proposition d'un nouveau modèle. Enfin nous expliquons comment nous avons essayé d'évaluer le domaine d'application de ce modèle

MOTIVATIONS

I - Présentation des systèmes séquentiels informatiques

II - Problèmes temporels posés par les systèmes séquentiels

III - Les modèles existants des systèmes séquentiels

IV - Le modèle proposé

V - Domaine d'application du modèle

MOTIVATIONS

Deux grandes théories permettent actuellement de formaliser la notion d'information. La première - la "théorie de l'information" - est relative à la transmission de l'information, elle a des fondements statistiques [SHA], [BRI]. *La seconde - la "théorie des langages* [CHO] *et des machines séquentielles"* [HUF], [MEA], [BOO] *- est relative au traitement de l'information et a des fondements algèbriques* [NOL].

Dans ces deux théories, la notion de temps n'apparaît pas sous la même forme. Dans la première, l'information a pour support des phénomènes caractérisés par des fonctions continues de la variable réelle "temps" [SIM]. *Par contre, dans la seconde théorie, la notion de temps se traduit par celle de séquence et la notion de durée pourrait se traduire par celle de longueur d'un mot.*

Différentes extensions de la théorie des machines séquentielles ont été développées afin d'éliminer cette différence de représentation - continue ou dénombrable - du temps [EVE], [SIF]. *Le modèle proposé dans cette thèse est aussi développé dans ce but, mais l'approche du problème diffère fondamentalement de celle qui est empruntée dans ces différentes extensions. Dans ce chapitre, nous essayons d'expliquer et de justifier le choix des fondements de notre modèle.*

I - PRESENTATION DES SYSTEMES SEQUENTIELS INFORMATIQUES

Nous nous limitons à l'étude des systèmes dont la fonction est le "traitement de l'information".

1.1. Traitement séquentiel de l'information

De la façon la plus générale, un système de traitement de l'information est un système auquel on fournit des *données* afin d'obtenir un *résultat*. Vu comme un système au sens des Sciences Physiques [LAD], un tel système est un système *ouvert* pour lequel les échanges (d'informations) avec l'extérieur se font dans les deux sens: en *entrée* (données) et en *sortie* (résultats).

Le traitement de l'information peut se faire:
- 1) *en une seule étape:* nous dirons que le système est *combinatoire;*
- 2) *en plusieurs étapes:* nous dirons que le système est *séquentiel.*

Théoriquement, tout traitement de l'information peut être effectué par un système combinatoire. Cependant, les systèmes séquentiels se justifient pour deux raisons différentes:
- 1) il est physiquement plus simple de décomposer un traitement complexe de l'information en une *suite* de traitements élémentaires et de réaliser la *chaîne* correspondante de systèmes séquentiels que de réaliser un système global fort complexe;
- 2) il existe des problèmes pour lesquels il est extrêmement difficile de fournir toutes les données "simultanément".

Pour les systèmes combinatoires, le traitement de l'information se compose d'un seul type d'étape:

données → *résultat* .

Par contre, un traitement séquentiel peut se décomposer en plusieurs sortes d'étapes:

	données	→	*résultat intermédiaire*	(1)
données +	*résultat intermédiaire*	→	*résultat intermédiaire*	(2)
	résultat intermédiaire	→	*résultat final*	(3)
données +	*résultat intermédiaire*	→	*résultat final*	(4)

1.2. Automate abstrait d'états

Un système séquentiel peut être considéré comme un *automate abstrait d'états* si l'on suppose:

- 1) que l'ensemble des étapes qui aboutissent à un résultat intermédiaire est l'*ensemble des états S*,
- 2) que l'ensemble des valeurs prises par les données est l'*alphabet d'entrée I*,
- 3) que l'ensemble sur lequel les résultats finaux prennent leur valeur est l'*alphabet de sortie O*.

Le fonctionnement de l'automate est alors caractérisé par les relations qui lient 1) la *suite** (séquence) des états de l'automate, 2) la suite des caractères émis en sortie et 3) la suite des caractères d'entrée.
Soient i_n , s_n *et* o_n les éléments génériques de ces suites. Avec ces conventions, un système séquentiel est caractérisé par un ensemble de relations des types suivants [HEB] :

Changement d'état: $i_n \in I, s_n, s_{n+1} \in S : \quad s_{n+1} = f(i_n , s_n)$ (5)

Ce type de relation correspond aux relations (1) et (2);

Sortie d'état :

a) Machine de Moore : $s_n \in S , o_n \in O : o_n = g(s_n)$ (6)

Cette relation correspond à la relation du type (3) ,

b) Machine de Mealy: $i_n \in I , s_n \in S , o_n \in O : o_n = h(i_n,s_n)$ (7)

Cette relation correspond à celle du type (4).

Notons qu'un système combinatoire pourrait être caractérisé comme un automate de ce type ayant plusieurs entrées (et par là plusieurs alphabets d'entrée I_1 , .. , I_j , .. , I_p) et un seul état:

$$i^1_n \in I_1 , i^j_n \in I_j , i^p_n \in I_p , o_n \in O : \quad o_n = k(i^1_n, .. , i^j_n, .. , i^p_n) \qquad (8)$$

Du point de vue de l'ordre séquentiel, ces relations correspondent aux relations (6) et (7).

* *Une suite est dénombrable par définition* [BOU]

1.3. Domaine d'application usuel de la théorie des machines séquentielles

Nous essaierons de définir ce domaine d'application de la façon suivante:
- soit un système S de traitement de l'information, caractérisé comme un automate d'états,
- soit un ensemble d'automates élémentaires $E = \{ E_i \}$ dont les alphabets d'entrée et de sortie sont (en général) binaires et qui peuvent correspondre à des systèmes élémentaires - ou bien combinatoires - ou bien séquentiels,
- alors le but de la théorie des machines séquentielles est de définir une structure composée d'éléments E_i *équivalente* au système S.

Il faut pour cela établir une correspondance bi-univoque :
- 1) entre l'alphabet d'entrée (resp. de sortie) de S et un ensemble-produit de *chaînes* de caractères des alphabets d'entrée (resp. de sortie) des éléments E_i;
- 2) entre l'ensemble des états de S et le produit des ensemble d'états des éléments E_i.

Le but de la théorie des machines séquentielles est naturellement d'arriver, à l'aide d'une telle décomposition, à une structure plus simple; c'est à dire [HEB] :
- soit à une structure d'éléments "logiques" d'un *catalogue* (eg: éléments combinatoires et cellules de mémoire) [*Dans ce cas, à chaque élément logique correspond un élément physique et à chaque entrée ou sortie correspond un signal*];
- soit à une *cascade* de systèmes séquentiels plus simples;
- soit à un *"produit"* de systèmes séquentiels (ce qui permet de remplacer un produit d'états par une *somme*: l'intérêt d'une telle décomposition est évident).

Après ce rappel de la notion de machine séquentielle, telle qu'elle est présentée dans de nombreux ouvrages, nous pouvons faire la remarque suivante: *la notion usuelle de machine séquentielle exclut toute notion de temps continu, car le temps n'apparaît que sous la forme de séquences dénombrables de caractères ou d'états.*

La solution du problème que nous soulevons semble tellement évidente qu'elle en est implicite. Ainsi, lorsque nous avons parlé de la décomposition d'un système séquentiel, nous n'avons pas abordé le problème de la correspondance du nombre de caractères des séquences d'entrée ou de sortie entre les

éléments E_i de la structure de synthèse. Implicitement cette correspondance est "unitaire" : à chaque caractère d'une séquence correspond un et un seul caractère des autres séquences.

Nulle part apparaît la notion de mesure *du temps.*

Autrement dit, le *repérage temporel* des entrées et des sorties est *unique* et prend en général la forme d'un *signal d'horloge h* . Celà implique que, physiquement, la mesure du temps est la même pour tous les signaux associés aux entrées et aux sorties des éléments d'une structure séquentielle.

Cette hypothèse unificatrice simplifie l'énoncé du problème, mais il n'est pas évident qu'elle n'entraîne pas des "paradoxes" semblables à celui " d'Achille et de la tortue" soulevé par Zénon d'Elée [*Nous pensons qu'à l'aide de ses quatre paradoxes, Zénon d'Elée voulait montrer qu'on ne peut considérer l'ensemble des nombres "réels" comme une simple extension de l'ensemble dénombrable des nombres rationnels*]. Nous allons maintenant essayer de répondre à cette question.

II - PROBLEMES TEMPORELS POSES PAR LES SYSTEMES SEQUENTIELS

2.1. Etablissement et maintien d'un état

Si un système séquentiel est considéré comme un système au sens des Sciences Physiques, la relation de changement d'état (5) : $s_{n+1} = f(i_n, s_n)$ peut être considérée comme la relation de *cause à effet* entre le caractère d'entrée i_n et le changement d'état $< s_n \rightarrow s_{n+1} >$ du système. L'ordre croissant des indices (n , n+1) traduit le *principe des effets retardés*, suivant lequel l'effet suit la cause.

Il ne faut pas alors oublier que pour un système "physique" *ouvert*, l'état de celui-ci dépend *à tout instant* (c'est à dire " quelle que soit la valeur de la variable *réelle t* ") de celui des entrées. Il en résulte que le symbole i_n de la relation (5) peut être interprété de deux façons distinctes:

- ou bien i_n caractérise l'*état* de l'entrée I, par rapport à un des deux états s_n ou s_{n+1} du système,
- ou bien i_n caractérise la *variation d'état* de l'entrée I qui est la

cause du changement d'état du système.

Dans le premier cas, le symbole i_n caractérise la valeur de l'entrée qui assure le *maintien* d'un état du système. Celà implique que la relation (5) n'est une relation de *cause à effet* que dans la mesure où les états de l'entrée I qui assurent le maintien de deux états *successifs* du système sont *distincts*.

Dans le second cas, le symbole i_n caractérise la *cause d'établissement* du nouvel état s_{n+1} du système. Cela implique que l'état de l'entrée I qui assure le maintien d'un état (s_n ou s_{n+1}) du système est *implicite* : cet état de l'entrée correspond à l'élément neutre / symbole séparateur que l'on rencontre dans les alphabets de la théorie des automates.

Cette double interprétation du symbole d'entrée i_n des relations de changement d'état (5) , due à l'hypothèse d'un repérage unique, entraîne la conclusion suivante:

> *la définition précédente des machines séquentielles ne peut aboutir à une théorie "unitaire".*

Nous entendons par là qu'il est impossible de combiner, à travers un seul et unique système de relations, d'une part un système séquentiel décrit comme une combinaison de systèmes combinatoires (systèmes "asynchrones") et d'autre part un système séquentiel décrit comme une combinaison d'éléments séquentiels (systèmes "synchrones").

Celà est dû au fait que les relations de cause à effet du type (8) ne peuvent s'appliquer aux systèmes asynchrones: nous nous heurtons alors à des contradictions, comme nous le démontrons dans la note 1 du chapitre III.

Cette impossibilité a provoqué le développement de deux théories distinctes : celle des *machines synchrones* [DIE] et celle des *machines asynchrones* [HEB].

Pour obtenir une théorie "unitaire", il faut nécessairement que la distinction entre l'établissement et le maintien d'un état apparaisse dans le système de relations. Par exemple, la relation de type (5) peut être remplacée par les suivantes:

établissement: $s_{n+1} = f(i_{2n+1} , s_n)$

maintien : $s_n = g(i_{2n} , s_n)$

Ces relations correspondent à celles de la théorie des automates, si l'on suppose que les symboles i_{2n} dénotent le symbole séparateur neutre e tel que $g(e , s_n) = s_n$.

2.2. Repérage des états

L' hypothèse d'unicité du repérage implique, d'un point de vue "physique", que le fait de caractériser l'état de tous les signaux du système à une *même* date *doit* avoir un sens. Nous allons d'abord montrer que,pour un système séquentiel physique,les effets et les causes ont une durée bornée *non nulle*. Nous rechercherons ensuite les conditions dans lesquelles le fait d'associer une date à un *intervalle* de temps a un sens.

a) Durée d'un état

La *longueur* de la séquence des états d'une machine séquentielle est *quelconque*. Cela implique, pour le système physique associé, que la valeur à une *date* T , d'une variable d'état $S(T)$ ne peut être fonction que des valeurs (des variables d'entrée I_j et d'état S_k) comprises dans des intervalles de temps *bornés* :

$$[t_j , T_j] \text{ ou } [t_k , T_k] \quad \text{avec} \quad t_j < T_j \leqslant T \; ; \; t_k < T_k \leqslant T$$

Borne supérieure : ces intervalles sont *majorés* , d'après l'axiome qui postule que l'effet suit la cause. Les intervalles peuvent être ouverts ou fermés à droite, celà dépend de l'interprétation stricte ou non du principe des effets retardés. Nous n'aborderons pas ce problème et renvoyons à Costa de Beauregard [COS] .

Borne inférieure : les intervalles sont *minorés* car une durée infinie impliquerait que l'état caractérisé par la valeur $S(T)$ ne peut être atteint , à partir d'un autre état, qu'au bout d'un temps infini. A la rigueur , seul l'état initial peut avoir cette propriété.

[*Cette minoration implique que le traitement "physique" de l'information est non linéaire , car seule la non linéarité permet "d'éliminer" le"passé"d'un signal*]

Durée non nulle : la correspondance entre une variable logique d'état - *entière* - et un signal - *fonction continue du temps* - ne peut être établie que si l'on suppose que la durée des effets et des causes n'est pas "nulle", c'est à dire si les effets et les causes ne sont pas "ponctuels" dans le temps.

La durée d'un état est donc bornée et non nulle.

b) Caractéristiques repérables d'un signal - support de l'information

Le repérage unique,d'un état dont la durée est bornée et non nulle, consiste à associer une seule et même date à *tous* les intervalles de temps du type $[\, t_j \, , \, T_j \,]$ ou $[\, t_k \, , \, T_k \,]$. Cette date peut correspondre :

- ou bien à une des deux bornes des intervalles de temps,
- ou bien à un "point" compris entre ces bornes.

Nous allons voir que seul le premier type de correspondance peut avoir une interprétation physique et cela au prix d'hypothèses supplémentaires.

Sur toute la durée d'un état, les valeurs des variables sont équivalentes par définition. Il en résulte qu'un"point" intermédiaire ne peut être repéré que par rapport aux *bornes* de l'intervalle de temps caractéristique de l'état [*Dans la pratique, cela revient à se repérer par rapport aux* transitions *d'un signal*]. Un tel repérage relatif n'a de sens que si toute transformation d'un signal est une *isométrie temporelle*, c'est à dire une application qui conserve la *distance* temporelle entre les bornes de repérage.

Or il s'avère que pour la plupart des éléments physiques dits " de commutation " (eg: logique TTL), les temps de retard " à la montée " et " à la descente " ne sont pas identiques. Cette propriété est indépendante de la fonction logique réalisée par l'élément physique. La propagation des signaux à travers les éléments logiques ne peut donc être considérée comme une isométrie temporelle.

Il en résulte que seules les transitions d'un signal peuvent servir de"point" de repère.

Pour que cette possibilité soit vérifiée, il faut s'assurer de plus que le temps de retard associable à une transition *ne dépende pas du signal*

(p.ex: de son temps de montée/descente)de façon à ce que toutes les transitions d'un même sens soit repérées de la même manière. Cette propriété d'invariance n'est pas vérifiable à l'aide des modèles traditionnels des éléments physiques. A notre connaissance, seuls les modèles de J. Suchard et D. Etiemble [SUC], [ETI] permettent de déterminer les conditions dans lesquelles cette invariance est vérifiée (cf. le préambule de l'annexe).

Dans ces conditions, la seule façon de caractériser un changement d'état à l'aide d'une date unique est :
- 1) de supposer que, avant de changer d'état, le système a atteint un état "stable", invariant dans le temps (cf. la définition de la stabilité [CAZ]);
- 2) de supposer que le début de la cause du changement d'état peut être repéré temporellement par une date unique;
- 3) de repérer temporellement le changement d'état par cette date.

Comme l'effet suit la cause et comme l'état initial est "stable", on peut en toute rigueur définir l'état du système *physique* à la date du *début* du changement d'état.

Cette hypothèse de stabilité est faite dans la théorie des machines synchrones : le repérage est effectué par rapport au *signal d'horloge* qui *commande* le changement d'état des éléments séquentiels (registres, compteurs, etc..) [*Notons que cette fonction de commande implique que la référence temporelle qu'est le signal d'horloge ait une existence matérielle*].

Par contre, dans la théorie des machines asynchrones, l'état d'un système est étudié en dehors des périodes de stabilité, bien que le repérage des signaux soit unique. L'interprétation "physique" des variablesassociées à l'état des signaux nous semble alors pour le moins délicate. C'est ce qui explique que dans de nombreux ouvrages les hypothèses suivantes sont ajoutées [CHU] :
- 1) les temps de retard sont indépendants du sens de la transition du signal;
- 2) tous les temps de retard ont une valeur multiple d'une même unité de temps.

Ces hypothèses ne nous semblent pas être des hypothèses "simplificatrices" qui seraient introduites afin d'alléger l'énoncé de la théorie : elles sont fondamentales. Autrement dit, nous ne pensons pas que la théorie des machines asynchrones puisse aboutir à une description fidèle d'un système *physique*

pour lequel les hypothèses précédentes ne sont pas vérifiées [*Ne serait-ce dû qu'à l'incertitude sur les caractéristiques physiques des éléments logiques, on ne peut dans la plupart des cas vérifier de telles hypothèses*].

2.3. "Déterminisme" des systèmes séquentiels

Pour terminer notre étude sur les problèmes temporels posés par les systèmes séquentiels, nous étudions l'influence de l'hypothèse du repérage unique sur le problème de l'évaluation du bon fonctionnement (*"déterminisme"*) d'un système séquentiel physique.

Nous caractériserons la finalité de la conception des ordinateurs de la façon suivante:

Synthèse : concevoir un système physique qui réalise le traitement de l'information demandé;

Analyse : définir les conditions physiques, en particulier temporelles, qui doivent être vérifiées pour que le système physique réalise effectivement le traitement demandé.

L'unicité du repérage ne permet qu'une formulation du but de l'analyse, celle de la vérification de l'*unicité du résultat*, relativement à un jeu de données d'entrée [*ie: il ne doit pas y avoir d' "aléas"*].

Comme il y a une infinité de valeurs possibles, pour les temps de propagation et de commutation et pour la durée des signaux, il y a aussi une infinité de cas à envisager. Il en résulte que cette façon d'énoncer les conditions de bon fonctionnement ne peut en général conduire à une réponse concrète [*Cette remarque ne s'applique pas à la théorie des machines synchrones, nous reviendrons sur ce sujet*].

La seule façon d'obtenir une réponse concrète, pour un système séquentiel quelconque, est de formuler la question "à l'envers", c'est à dire sous la forme suivante:

Pour un résultat logique et un jeu de données d'entrée imposés et un système physique donné,

- *1) existe-t-il des cas de bon fonctionnement?*
- *2) si oui, par rapport à quel repère temporel peut-on observer,dans chaque cas, le résultat et est -ce que ce repère convient?*
- *3) quelles sont alors les contraintes temporelles que doivent respecter les signaux de données, relativement à ce repère?*

Seule cette formulation permet d'apporter une réponse exhaustive sur le "déterminisme" d'un système séquentiel physique.

2.4. Influence de l'hypothèse de repérage unique

L'étude précédente a mis en évidence certaines conséquences du choix de l'hypothèse d'un repérage unique :
- 1) ce choix ne permet pas de décrire la totalité des sytèmes séquentiels à l'aide d'un seul et unique système de relations, compatible avec la théorie des automates abstraits;
- 2) la correspondance entre un état du système séquentiel de départ et un état du système physique associé ne peut être établi que pour un sous-ensemble restreint des systèmes séquentiels (ceux de la théorie des machines synchrones);
- 3) à l'exception des systèmes séquentiels "synchrones", ce choix ne permet pas de déterminer les conditions limites de bon fonctionnement du système physique associé au système séquentiel de départ.

Nous essayons maintenant de caractériser les modèles existants de machines séquentielles, face à ce problème de repérage temporel.

III - LES MODELES EXISTANTS DES SYSTEMES SEQUENTIELS

D'après le résultat de nos recherches bibliographiques, nous avons classé les modèles existants en trois catégories:

1) Théorie des machines asynchrones,
2) Théorie des machines synchrones,
3) Logiques booléennes temporelles.

A ces trois catégories, il faudrait ajouter les modèles utilisés en simulation qui ne peuvent être rattachés à une des catégories précédentes. Nous les avons exclu pour les raisons suivantes.

3.1. Modèles de simulation

Certains modèles correspondent à une simulation "électronique" des éléments séquentiels. Cette caractérisation purement physique des systèmes informatiques exclut toute notion d'automate d'états, c'est à dire de traitement logique de l'information. Si ce modèle de simulation permet d'étudier le fonctionnement électromagnétique du système, nous ne pensons pas qu'il permette d'en étudier le fonctionnement logique.

D'autres modèles remplacent les signaux binaires discontinus " en créneaux < 0 , 1 > " par des signaux qui possèdent des " rampes " de transition. L'état du signal est alors partout défini et prend toutes les valeurs comprises entre celles qui caractérisent les deux états "logiques" du signal. Cette finesse de la simulation nous semble plus qu'illusoire , car nous avons vu que la notion de date, c'est à dire de point auquel on attache une valeur du signal, n'a guère de sens [*Sauf pour les dates qui caractérisent une transition du signal*].

3.2. Théorie des machines asynchrones

Nous avons déjà exposé les limitations temporelles de cette théorie. Cette théorie permet d'effectuer la synthèse d'un système séquentiel en termes d'éléments combinatoires, c'est à dire en termes d'éléments logiques de base. Elle permet d'étudier le comportement temporel du système, lorsque les valeurs des temps de retard , ainsi que les caractéristiques temporelles des signaux, sont donnés. Elle ne permet pas cependant de déterminer l'ensemble des contraintes temporelles qui assurent le bon fonctionnement du système.

3.3. Théorie des machines synchrones

Cette théorie nous semble la plus complète: elle répond à la finalité de la conception des ordinateurs, telle que nous l'avons définie.

Pour qu'un système physique conçu selon les principes de cette théorie fonctionne, il suffit en effet que les conditions de stabilité des états soient vérifiées avant chaque changement d'état. Ces conditions de stabilité se résument aux deux contraintes suivantes:

- 1) l'intervalle de temps qui sépare deux changements d'état doit être supérieur à la somme du plus grand temps de commutation des éléments séquentiels de base et du plus grand temps de propagation d'un signal entre deux éléments séquentiels de base;

- 2) les éléments séquentiels de base doivent être *non transparents* [*ie: le changement d'état doit se faire en deux étapes : a) lecture des entrées, b) "écriture" des sorties*].

Ces contraintes peuvent être respectées *à l'intérieur* du système, puisque tous les éléments séquentiels de base sont commandés par un même signal d'horloge. Par contre, pour les signaux d'entrée du système, le respect de ces contraintes implique que le "monde extérieur" au système soit synchrone avec celui-ci ! Le domaine d'application de la théorie des machines synchrone est donc limité, puisque les systèmes combinatoires- ainsi que les systèmes qui comportent plusieurs références de temps - en sont exclus. Cependant, l'utilité de cette théorie ne peut être niée, mais il faut bien reconnaître qu'elle se rapproche beaucoup plus de la théorie des automates que de la conception physique des ordinateurs !

3.4. Les logiques booléennes temporelles

Cette théorie est très intéressante car elle représente une tentative d'élimination de l'hypothèse du repérage unique [SIF].

Chaque signal est repéré de façon intrinsèque et est représenté sous la forme d'une somme booléenne d' "échelons" dont les fronts repèrent les transitions du signal. Cette représentation pourrait être comparée à la décomposition d'un signal en séries de Fourier [MAX]. Cette approche est nécessaire pour étudier un système combinatoire comme un système physique [*Seules les transitions d'un signal peuvent servir de point de repère*],mais elle ne permet pas d'étudier le système comme un automate d'états.

Le système de repérage proposé ne permet pas en effet d'observer les transformations successives de l'état *logique* du système. Cette impossibilité résulte de l'*unicité de l'origine temporelle* pour tous les signaux, alors que la date des transitions qui délimitent un état du système n'est pas la même

pour chaque signal. Il est alors impossible d' "isoler" temporellement un état, car les lois booléennes de composition des signaux impliquent que tous les signaux doivent être *définis* pour être combinés : leur origine temporelle (d'observation) ne peut être qu'identique.

Pour "observer" un état isolément, il faut *dissocier l'état et le repérage d'un signal*. Seules des lois de composition distinctes pour l'état et pour le "temps" permettent d'associer une date spécifique d'observation à chaque signal, tout en conservant les lois de composition booléennes pour l'état des signaux.

IV - LE MODELE PROPOSE

Le modèle que nous proposons a été conçu dans le but de concilier deux types de représentation d'un système séquentiel :

- la représentation d'un système comme un automate abstrait d'états,
- la représentation d'un système comme un système physique pour lequel le temps est continu.

Pour cela, le modèle est fondé sur la *DUALITE DES NOTIONS D'ETAT ET DE TEMPS*. Nous considérons en effet que , bien qu'elles soient complémentaires - et par là indissociables - , ces deux notions sont *distinctes :*

- la notion d'état d'un signal n'a de sens que par rapport à une référence de temps et
- réciproquement une référence de temps ne peut être définie que par l'état d'un signal,
- cependant l'état et la référence de temps *ne sont pas obligatoirement associés au même signal :* ces deux notions sont relatives à un système de repérage qui peut être quelconque.

Cette dualité est la cause du développement d'un formalisme spécifique à chaque notion.

4.1. Logique d'état

La logique booléenne traditionnelle a été remplacée par une logique " extrêmale" pour les raisons suivantes:

- 1) Dans certains traitements de l'information, les alphabets associés aux signaux ne sont pas identiques; en particulier, les alphabets associés

aux signaux de commande sont en général binaires alors qu'il n'en est pas toujours de même pour les signaux de données [*Rappelons que les lois de l'algèbre de Boole sont des lois de composition internes : il ne peut alors y avoir qu'un seul alphabet*].

- 2) La distinction entre l'*établissement* et le *maintien* d'un état doit être mise en évidence : c'est le rôle des *extremums* des fonctions extrêmales.
- a) Si une des entrées a la valeur *maximale*, alors la valeur de sortie de la fonction est imposée, quelle que soit la valeur des autres entrées. Cette propriété caractérise une fonction d'*initialisation* , c'est à dire d'établissement.
- b) Par contre, une valeur *minimale* d'entrée n'influe pas sur la valeur de sortie de la fonction, valeur qui ne dépend alors que des autres entrées. La valeur minimale peut être considérée comme un caractère *séparateur* (élément neutre) qui"maintient" la valeur de la fonction, puisqu'elle ne la modifie pas.

A chacune des deux lois d'union et d'intersection de l'algèbre de Boole correspond un élément "maximal" et un élément "minimal" [*Union: élément maximal 1 , minimal 0 ; Intersection : élément maximal 0 , minimal 1*].Ces deux lois sont remplacées par une seule fonction extrêmale, lorsqu'il s'agit de caractériser l'établissement ou le maintien d'un état.

4.2. Logique temporelle

Comme nous l'avons vu, il faut que l'origine temporelle d'un signal ne soit pas un "invariant" de la logique temporelle. C'est pourquoi cette logique est fondée sur une relation d'*inclusion* temporelle *partielle*. La relation d'inclusion ne porte que sur une partie du signal, partie qui est définie comme l'ensemble des composantes qui correspondent à une *partie* de l'alphabet associé. Une composante est alors *incluse* temporellement dans une autre si elle débute *après* et finit *avant*.

Les lois d'*union* et d'*intersection* sont construites sur cette relation d'inclusion. Leur particularité est de "modifier" la date d'origine qui ne peut que *progresser* dans le temps. Cette *irréversibilité* temporelle est nécessaire pour l'étude des phénomènes d'initialisation, cependant elle

implique que l'algèbre résultante ne peut être une algèbre de Boole.

L'algèbre est complétée par une loi de "complémentation" temporelle: la *mémorisation*. Cette loi caractérise la fonction de base des systèmes séquentiels qui est celle de *décalage temporel* (ie: passage d'un indice *n* à l'indice *n+1*). Comme nous le montrons dans la note 2 du chapitre II, cette fonction de mémorisation mérite bien le nom de complémentation temporelle, malgré son apparence.

4.3. Les signaux comme éléments de base

Les deux algèbres proposées s'appliquent à des ensembles de signaux qui sont caractérisés de deux façons distinctes:

- 1) à l'aide d'*alphabets* que nous distinguons soigneusement de l'ensemble des valeurs prises par la grandeur caractéristique du signal, car - rappelons le - la notion d'état est relative à un système de repérage : elle dépend de ce que l'on observe.
- 2) à l'aide de *repères* partiels ou complets.

Nous avons porté un soin tout particulier à la définition des alphabets et des repères, afin de bien mettre en apparence la *relativité* de ces notions. Nous montrons en particulier que la mesure de l'information *dépend de ces définitions* [*On peut le vérifier pour les enregistrements magnétiques: la référence de temps ("synchro") peut être intégrée au signal ou bien peut être enregistrée séparément; le dénombrement des configurations distinctes du signal diffère dans chaque cas. Cette notion de mesure relative à la dualité " état - temps " se retrouve dans les relations d' Heisenberg* LEV].

4.4. Schéma d'analyse

La définition d'une algèbre, même double, n'est pas suffisante pour établir une théorie:il faut de plus définir des règles de construction [BOU]. [*Pour l'analyse d'une structure, il s'agirait plutôt de règles de décomposition*] Pour celà nous avons fait appel à la théorie des langages: nous caractérisons les possibilités de fonctionnement d'un système séquentiel comme les *phrases* du langage qui est "accepté" par le système. Cette approche a été choisie car elle permet de répondre à la question du "déterminisme" (ie: bon fonctionnement)

du système physique correspondant au système séquentiel.

La méthode proposée est semblable à celle que l'on peut utiliser pour évaluer la "puissance" d'un automate. Cette puissance est évaluée en vérifiant que l'automate accepte un langage d'un certain type. Il faut pour cela:
- 1) vérifier la compatibilité des vocabulaires,
- 2) vérifier que l'automate accepte les différentes structure de phrases du langage.

Dans notre modèle, les "éléments du vocabulaire" sont les états élémentaires du système. A l'opposé de la théorie des automates, où les états sont distingués des entrées, nous caractérisons un état par un couple d'*isochrones* < *causes* , *effet* > [*Une isochrone est un ensemble de composantes de signaux muni d'un système de repérage*], car la continuité du temps physique ne permet pas de dissocier temporellement les causes des effets (à tout instant il y a une cause pour un effet observé).

L' "acceptation" d'un état élémentaire consiste alors à vérifier :
- 1) que les causes indiquées impliquent bien l'effet donné,
- 2) que cette relation de cause à effet peut effectivement être observée par rapport au système de repérage donné (au prix de certaines contraintes temporelles qui sont alors déterminées).

Les "phrases" du langage sont obtenues par *concaténation* d'états élémentaires du "vocabulaire" [*Nous avons pour cela"adapté" la loi de concaténation des algèbres monoïdes à des ensembles réels*]. L' acceptation des phrases se traduit par les vérifications suivantes :
- 1) les signaux de sortie doivent pouvoir être *complètement* reconstitués à partir de la concaténation des isochrones-causes,
- 2) la concaténation des isochrones-effets doit être temporellement compatible avec une *succession* des isochrones-causes associées [*la succession des isochrones-causes - et non pas la concaténation - correspond à la recherche des causes minimales*].

Ces vérifications permettent d'affirmer que le bon fonctionnement du système est *observable* relativement à un certain système de repérage; encore faut-il que ce système de repérage ait un intérêt logique [*Par exemple, un*

repérage relatif aux sorties - c'est à dire au résultat d'un traitement de l'information - est possible, mais il n'a aucun intérêt puisque ce résultat est par hypothèse inconnu].

La puissance de cette méthode - issue de la théorie des automates - permet d'étendre l'analyse de la structure à un système pour lequel l'incertitude sur la valeur des paramètres temporels n'est pas nulle.

Nous sommes parti de la remarque que - à l'intérieur d'une machine - la définition du temps est univoque (sauf à l'intérieur de l'horloge) et que tout système de repérage a alors une structure *arborescente* dont l'horloge est la racine. La méthode précédente est appliquée à partir des "feuilles" pour remonter jusqu'à la racine en cumulant progressivement les incertitudes. Cette technique de remontée correspond à celles de l'analyse syntaxique, lors de la compilation d'un langage de programmation: *la méthode proposée est automatisable.*

V - DOMAINE D'APPLICATION DU MODELE

Au lieu de délimiter précisément le domaine d'application de notre modèle , nous essayons plutôt de montrer,à travers des applications tirées de la pratique - et par là fort bien connues -,l'intérêt
- 1) que l'on retire de l'abandon de l'hypothèse du repérage unique,
- 2) d'une classification "temporelle" des systèmes séquentiels.

Nous pensons en effet que les problèmes spécifiques à la conception des systèmes séquentiels informatiques sont surtout temporels; les autres problèmes relèvent d'autres domaines: - conception des éléments logiques,
- conception des automates abstraits d'états.

L' intérêt d'une classification temporelle est alors d'isoler chaque problème spécifique aux systèmes séquentiels , afin de le résoudre *très simplement.*

Nous proposons dans ce but la classification temporelle suivante:

a) Les systèmes combinatoires

Ces systèmes ne possèdent pas de système de repérage intrinsèque et n'importe quel signal d'entrée peut servir de référence temporelle [*cela*

n'implique pas que le système de repérage soit identique pour chaque entrée]. Pour ces systèmes, l'état et le repérage des sorties peuvent être déduits des entrées.

b) Les systèmes à mémoire

Comme les précédents, ces systèmes ne possèdent pas de système de repérage intrinsèque, mais un seul signal d'entrée peut être utilisé comme référence de temps: *la commande d'écriture*. Seul le repérage des sorties peut être déduit de l'observation de cette entrée, car l'état des sorties dépend des états précédents du système. Ces systèmes sont caractérisés physiquement par un *système d'échantillonnage* des entrées,commandé par le signal de commande d'écriture.

c) Les systèmes à temps propre

Ces systèmes possèdent leur propre système de repérage. Cela implique que ni l'état ni le repérage des sorties ne peut être déduit de l'observation des entrées. Ces systèmes possèdent deux types de dispositifs:

- une horloge,
- des mécanismes de *couplage* et de *découplage* temporel, afin de communiquer avec d'autres systèmes à temps propre.

Nous classons les éléments *monostables* dans cette catégorie : nous les considérons comme les constituants de base d'une horloge.

Cette classification nous permet d'isoler les problèmes temporels suivants
systèmes combinatoires: définition d'un système de repérage;
systèmes à mémoire: mémorisation, échantillonnage et transparence;
systèmes à temps propre: définition d'un système de repérage à références multiples et couplage / découplage temporel.

Afin de profiter d'exemples concrets, nous avons réparti l'exposé de la méthode d'évaluation(du "déterminisme" d'un système séquentiel)parmi les différentes analyses de problèmes temporels. Ainsi :

- 1) le report des contraintes temporelles de la sortie vers les entrées d'un élément logique est exposé à propos des systèmes combinatoires;
- 2) l' "acceptation" d'un état élémentaire et d'une suite de tels

états est présentée lors de l'étude des systèmes à mémoire. La méthode décrite ne tient pas compte des incertitudes et est rapportée au signal de commande d'écriture. La détection d' "aléas" est illustrée à l'aide de plusieurs exemples;

- 3) enfin, l'algorithme complet d'acceptation , qui tient compte des incertitudes et qui se rapporte à un système de références multiples, est décrit à propos des systèmes à temps propre.

Le plan de la thèse résulte de cette présentation des problèmes: après un chapitre où nous présentons de façon formelle notre modèle, nous étudions successivement son application aux systèmes à mémoire,puis aux systèmes à temps propre; l'application aux systèmes combinatoires est renvoyée en annexe.

CHAPITRE I I

Ce chapitre définit formellement les principes fondamentaux du modèle. Ceci nous amène tout d'abord à définir un schéma de construction d'une structure monoïde à partir d'un ensemble de signaux / fonctions continues du temps. Dans la deuxième partie, la notion de repère d'un signal , déduite de la structure monoïde, conduit à l'évaluation d'une mesure relative de l'information et au développement d'une algèbre temporelle. Ensuite, l'algèbre duale "statique" est caractérisée à l'aide de fonctions extrêmales. Enfin, la notion d'isochrone permet de définir un système séquentiel "physique" comme un automate d'états.

LE MODELE PROPOSE

I - La concaténation
II - Les repères
III - Algèbre des repères
IV - Les fonctions extrêmales
V - Les isochrones

I - LA CONCATENATION

Dans la Théorie des langages, un *vocabulaire* est défini comme un ensemble de suites (*mots*) dont les éléments sont les caractères de l'alphabet A du langage. La notion de langage implique que cet alphabet A est *fini* et que la "longueur" des mots est pour le moins *dénombrable* si ce n'est finie. Autrement défini, un vocabulaire est un sous-ensemble de l'ensemble M des mots que l'on peut construire sur l'alphabet A à l'aide d'une loi de composition associative qui possède un élément neutre: la *concaténation*. Cette loi définit sur l'ensemble M une algèbre dite monoïde.

Bien qu'il ait " la puissance du continu " , un ensemble de signaux, qui sont des fonctions continues du temps, peut être muni d'une algèbre de ce type : il suffit pour celà de définir un élément neutre et une loi de concaténation.

Dans ce but, chaque signal $S(t)$ est caractérisé par sa *durée* D et son *domaine de variation* V (ie: l'ensemble des valeurs prises en fonction du temps t), c'est à dire par un *pavé* [V x D]. Tout signal peut en ces termes être décomposé d'une infinité non dénombrable de façons en une succession de pavés de ce type.

La concaténation est alors définie comme la règle de construction d'un signal - défini lui même comme un pavé [V x D] - à l'aide de pavés élémentaires.

Tout signal dont le domaine de variation ou bien dont la durée est nulle est élément neutre de cette loi de composition.

L'ensemble des pavés élémentaires constitue l' "alphabet" du monoïde. Cet "alphabet" n'est pas dénombrable, puisque la durée et le domaine de variation de chaque pavé sont des intervalles de nombres réels. Autrement dit, à la différence du monoïde M précédent, l'algèbre est *sans atome*. La notion de longueur exprimée comme le nombre de caractères d'un mot n'a

alors plus de sens. Elle doit être remplacée par celle de *durée* ou, plus exactement, par l'application qui fait correspondre à la durée d le segment $[\ 0,\ d\ ]$ de l'ensemble des Réels positifs.

Dans la partie qui suit, cette algèbre monoïde est définie formellement. L' ensemble des signaux générés par concaténation de pavés $[\ V \times D\ ]$ est un peu plus vaste que celui des fonctions continues du temps, c'est pourquoi nous définirons le sous-ensemble des signaux continus.

Nous noterons $M(V,D)$ l'ensemble de tous les signaux dont la durée est inférieure ou égale au nombre *réel* D et dont le domaine de variation est inclus dans l'ensemble *ordonné* V, équipotent à l'ensemble des nombres réels. D'autre part, nous noterons $m(v,d)$ un élément de $M(V,D)$ de durée d et de domaine de variation v. Nous désignerons par $v(m)$ le domaine de variation de m et par $d(m)$ sa durée.

1.1. Définitions préliminaires

a) Jonction sur un ensemble ordonné

Soit E un ensemble ordonné et $I(E)$ l'ensemble des intervalles non vides définissables sur E . Soit ϕ l'intervalle vide et $J(E)$ l'ensemble $\{\ I(E)\ ,\ \phi\ \}$. La *jonction* , notée $-$, est une application de $J(E)$ dans $J(E)$ qui a les propriétés suivantes :

1) associativité : $\forall\ \alpha, \beta, \gamma \in J(E)$: $\alpha - (\beta - \gamma) = (\alpha - \beta) - \gamma$

2) commutativité : $\forall\ \alpha, \beta \in J(E)$: $\alpha - \beta = \beta - \alpha$

3) élément neutre: $\forall\ \alpha \in J(E)$: $\alpha - \emptyset = \emptyset - \alpha = \alpha$

4) Si $(inf(I), sup(I))$ et $(inf(J), sup(J))$ sont les bornes de deux intervalles non vides de $I(E)$, alors les bornes de l'intervalle $I - J$ sont égales à :

$$inf(I - J) = min(inf(I), inf(J))$$

$$sup(I - J) = max(sup(I), sup(J))$$

5) Soit une classe {I,J} de deux intervalles de I(E) . Soit λ et μ les éléments de cette classe tels que :

$$inf(\lambda) = min(inf(I), inf(J))$$
$$sup(\mu) = max(sup(I), sup(J)) .$$

Alors, le type de l'intervalle I - J est défini de la manière suivante :

a) λ *unique :*

λ ouvert (resp. fermé) à gauche $\Rightarrow$ I - J ouvert (resp. fermé) à gauche ;

b) $inf(I) = inf(J)$

Si I et J sont ouverts à gauche alors I - J est ouvert à gauche,
sinon I - J est fermé à gauche ;

c) μ *unique*

Si μ est ouvert (resp. fermé) à droite alors I - J est ouvert (resp. fermé) à droite ;

d) $sup(I) = sup(J)$

Si I et J sont ouverts à droite alors I - J est ouvert à droite,
sinon I - J est fermé à droite.

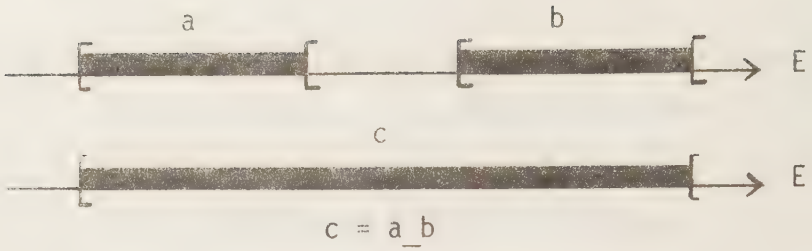
b) Ensemble M(V,D)

Soit le domaine de variation V un ensemble ordonné équipotent à l'ensemble des réels et soit I(V) l'ensemble des intervalles non vides définissables sur V. Soit la durée D un nombre réel et soit I(D) l'ensemble des segments de nombres réels $[0,d[$ (notés d) tels que $d \leq D$.

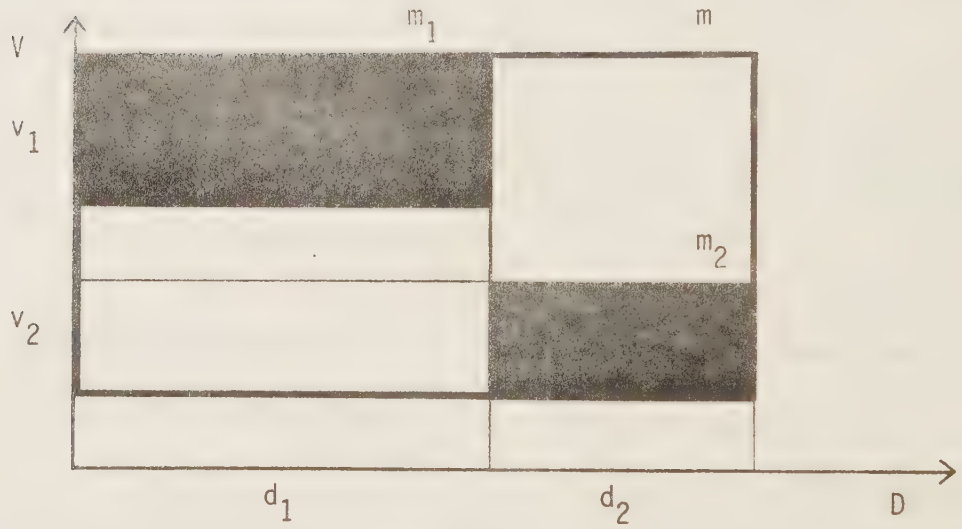
Par définition de l'ensemble M(V,D) , il existe une application f qui, à tout élément de I(V) x I(D) fait correspondre une partie de M(V,D) telle que : - *1)* le noyau de l'application est constitué des éléments dont une des projections est vide :

a) $f(\phi,d) = \phi \quad \forall\, d \in I(D)$; *b)* $f(i,\phi) = \phi \quad \forall\, i \in I(V)$

- *2)* deux éléments distincts de I(V) x I(D) qui n'appartiennent pas au noyau ont des images disjointes.

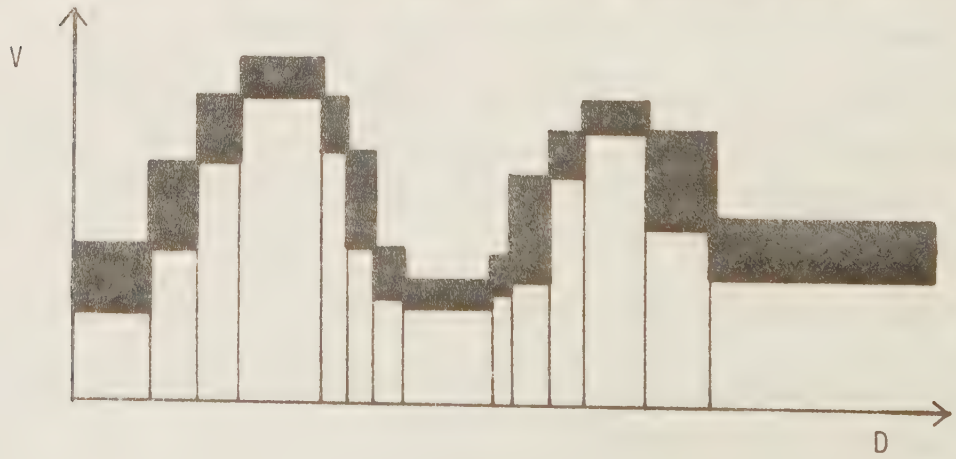


Jonction de deux intervalles a et b d'un ensemble ordonné E



$m = m_1 \hat{} m_2$

Concaténation m de deux signaux m_1 et m_2 (pavés de M(V,D))



"Signal" m(v,d), ie: élément de l'ensemble M(V,D)

1.2. Loi de concaténation

L'ensemble $M(V,D)$ est muni d'une loi de composition interne notée $\frown$: la concaténation. Cette application de $M^2(V,D)$ dans $M(V,D)$ possède par définition les propriétés suivantes :

1) associativité : $\forall \alpha,\beta,\gamma \in M(V,D) : (\alpha \frown \beta) \frown \gamma = \alpha \frown (\beta \frown \gamma)$

2) élément neutre : $\forall \alpha \in M(V,D) : \alpha \frown \phi = \phi \frown \alpha = \alpha$

3) domaine de variation : La concaténation des deux signaux entraîne la jonction de leurs domaines de variation :

$$\forall\ \alpha\ ,\ \beta \in M(V,D)\ \ : v(\alpha \frown \beta) = v(\alpha) - v(\beta)$$

4) la durée : la durée de deux signaux concaténés est égale à la somme de leurs durées :

$$\forall \alpha,\beta \in M(V,D) : \ d(\alpha \frown \beta) = d(\alpha) + d(\beta)$$

5) Complétude temporelle : tout signal est décomposable d'une infinité de façons :

$$\forall\ m(v,d) \in M(V,D)\ ,\ \forall\ e \leq d\ \ ,\ \exists\ \alpha,\beta \in M(V,D)\ :$$

$$m = \alpha \frown \beta$$

$$\text{et}\quad d(\alpha) = e$$

6) égalité : par rapport à la durée, la décomposition d'un signal est cependant unique :

$$\forall\ m\ ,\ \alpha,\beta,\gamma,\delta \in M(V,D) : m = \alpha \frown \beta \quad \text{et} \quad m = \gamma \frown \delta \quad \text{et} \quad d(\alpha) = d(\gamma)$$

alors : $\alpha = \gamma$ et $\beta = \delta$.

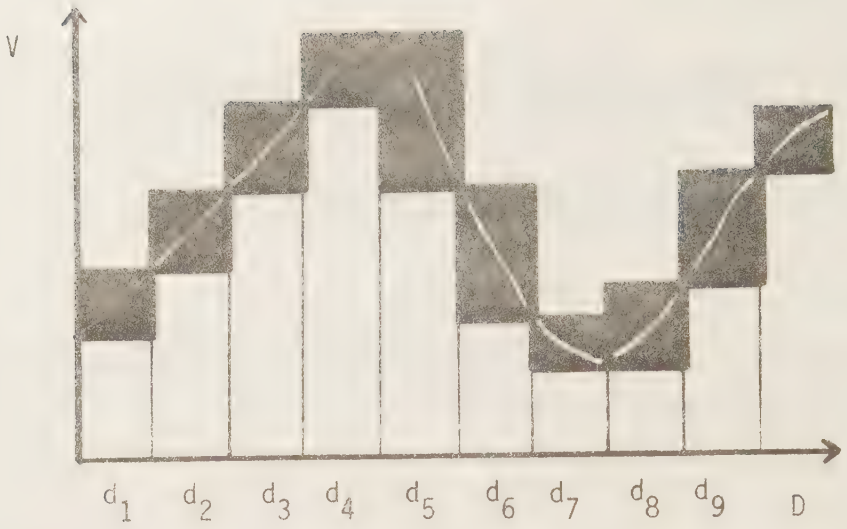
De cette définition de la concaténation découle une propriété caractéristique des monoïdes :

a,b,c,d étant des signaux de $M(V,D)$, si l'on a : $a \frown b = c \frown d$, alors il existe un signal h de $M(V,D)$ tel que :

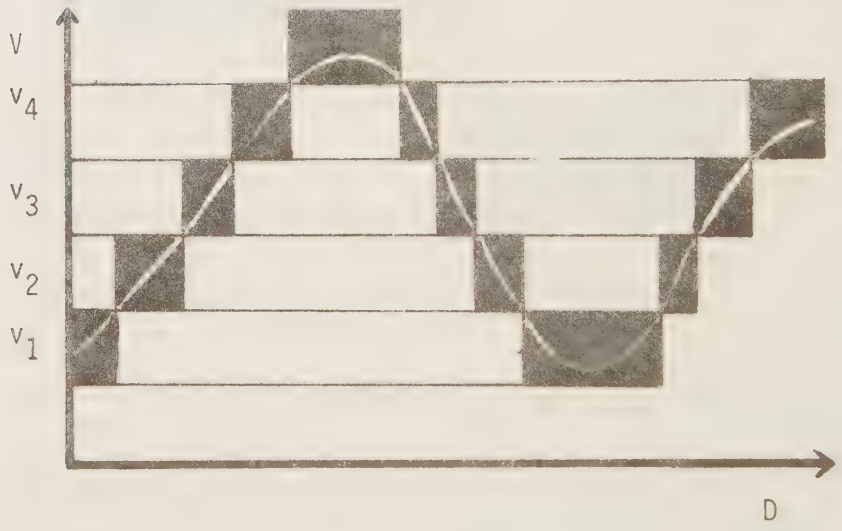
Soit $c \frown h = a$ et $h \frown b = d$,

Soit $a \frown h = c$ et $h \frown d = b$.

Pour le montrer, envisageons les trois cas possibles :



1) Repèrage relatif à l'échelle de mesure des durées D



2) Repèrage relatif à l'échelle de mesure du domaine de variation V

Exemple de deux signaux de M(V,D) qui "enveloppent" un signal continu

1) $a = c$ alors *(6)* : $b = d$ et $h = \emptyset$

2) $d(a) < d(c)$. D'après (5), il existe α , h tels que $d(\alpha) = d(a)$ et $c = \alpha^{\frown}h$. Comme $a^{\frown}b = c^{\frown}d$, $a^{\frown}b = \alpha^{\frown}h^{\frown}d$ et d'après (6) $a = \alpha$ car $d(a)=d(\alpha)$. Il en résulte que $a^{\frown}b = a^{\frown}h^{\frown}d$. D'après (1) $a^{\frown}b = a^{\frown}(h^{\frown}d)$ et d'après (6) $b = h^{\frown}d$.

3) $d(a) > d(c)$: la démonstration est analogue à la précédente.

1.3. Signaux continus

a) Définitions préliminaires

Composante : un signal b de $M(V,D)$ est une composante d'un signal m de $M(V,D)$ si il existe un couple de signaux (a,c) tels que m soit égal à a-b-c. L'abscisse de la composante b , par rapport à m, est alors par définition, la durée du signal a .

Amplitude : Seul le modèle physique détermine le type du domaine de variation des signaux : intervalle ouvert, fermé ou semi-ouvert . Il est donc difficile de définir une mesure du domaine de variation. Cependant, comme un domaine de variation est équipotent à l'ensemble ordonné des nombres réels, il existe toujours deux intervalles semi-ouverts de $V(M)$ dans lesquels le domaine de variation est emboîté. Cette propriété peut s'exprimer formellement ainsi :

$$\forall\, m \in M(V,D)\ ,\ \exists\, M_i\ ,\ M_\delta\ ,\ m_i\ ,\ m_\delta \in V \quad \text{tels que :}$$

$$[m_i\ ,\ m_\delta[\subseteq v(m) \subseteq [M_i\ ,\ M_\delta\ [$$

Si l'on définit la mesure d'un segment semi-ouvert $[inf,sup[$ par la différence sup-inf, il est possible de caractériser le domaine de variation $v(m)$ par le couple $(m_\delta - m_i\ ,\ M_\delta - M_i)$. Par définition, nous disons que l'amplitude $a(m)$ du signal $m(v,d)$ est bornée par ce couple. L'amplitude d'un signal n'est une distance que lorsque le domaine de variation est un segment semi-ouvert.

b) Continuité

Avec cette terminologie, un signal m de $M(V,D)$ est dit continu sur $[0,D[$ si à tout couple de nombres positifs réels (ε,e) on sait faire correspondre un nombre positif η , tel que toute composante de $m(v,d)$ d'abscisse e et de durée inférieure à η a de ce fait une amplitude inférieure à ε . Ce que nous écrivons en abrégé :

$$\forall\ \varepsilon\ ,\ e > 0, \exists \eta > 0 \text{ tel que } \left\{ \begin{array}{l} m = \alpha\,\beta\,\gamma \\ d(\alpha) = e \\ d(\beta) < \eta \\ v(\beta) \neq 0 \end{array} \right\} \Rightarrow a(\beta) < \varepsilon$$

c) Continuité uniforme

Nous définirons de la même façon un signal m uniformément continu sur $d(m)$ et à valeurs dans $v(m)$:

$$\forall \varepsilon > 0\ ,\ \exists \eta > 0 \text{ tel que } \left\{ \begin{array}{l} \forall e \leq d(m) \\ m = \alpha\,\beta\,\gamma \\ d(\alpha) = e \\ d(\beta) < \eta \\ v(\beta) \neq 0 \end{array} \right\} \Rightarrow a(\beta) < \varepsilon$$

d) Chemin de transition

Si une partition P est définie sur l'ensemble des composantes M(V,D), celà implique que les différentes parties de P recouvrent le domaine de variation V . Il en résulte, pour les signaux continus, que la transition entre deux composantes a et b , dont les domaines de variation ne sont pas adjacents, traverse obligatoirement au moins une troisième partie de P . Ce que nous écrivons formellement ainsi :

Soit P une partition M(V,D) , P_a et P_b deux parties telles qu'il existe un domaine de variation v_c - inclus dans la jonction des domaines v_a et v_b des parties P_a et P_b - et d'intersection nulle avec les domaines v_a et v_b: il existe alors un sous ensemble P_c de parties de P , distinctes de P_a et P_b , dont la jonction des domaines inclut v_c .

Dans ces conditions, si un signal $m(v,d)$ possède une composante a qui appartient à P_a et une autre b à P_b , alors il possède aussi :

- 1) une composante c non nulle qui appartient à P_c ,
- 2) deux composantes x et y telles que :

$$m(v,d) = x\,(a\,c\,b)\,y\ ,\ c \neq \emptyset \quad \text{et} \quad c \in P_c\ .$$

1.4. Origine d'un signal

Pour qu'il puissent être comparés entre eux, les signaux doivent être repérés par rapport à une même référence de temps absolu. C'est pourquoi nous étendons l'algèbre précédente afin d'introduire la notion d'origine.

a) Signal daté

Soit M(V,D) un ensemble de signaux et soit T un ensemble isomorphe à l'ensemble des réels $\mathbb{R}$. Un signal daté δ est alors un élément (t,m) de l'ensemble produit $S = T \times M(V,D)$, où t est l'origine du signal δ .

L'élément vide de $S = T \times M(V,D)$ est tout élément vide de $M(V,D)$

$$\emptyset_S : (t,\emptyset_{M(V,D)}) \quad \forall\, t \in T \ .$$

b) Changement d'origine

Par application du principe des effets retardés, nous définirons un changement d'origine comme une application T_x de $S \times T$ dans S qui ne possède pas d'application inverse. Ses propriétés sont les suivantes :

1) $T_x(\emptyset) = \emptyset$

2) $\forall_\delta \in S$, $T_x(\delta) \in S$

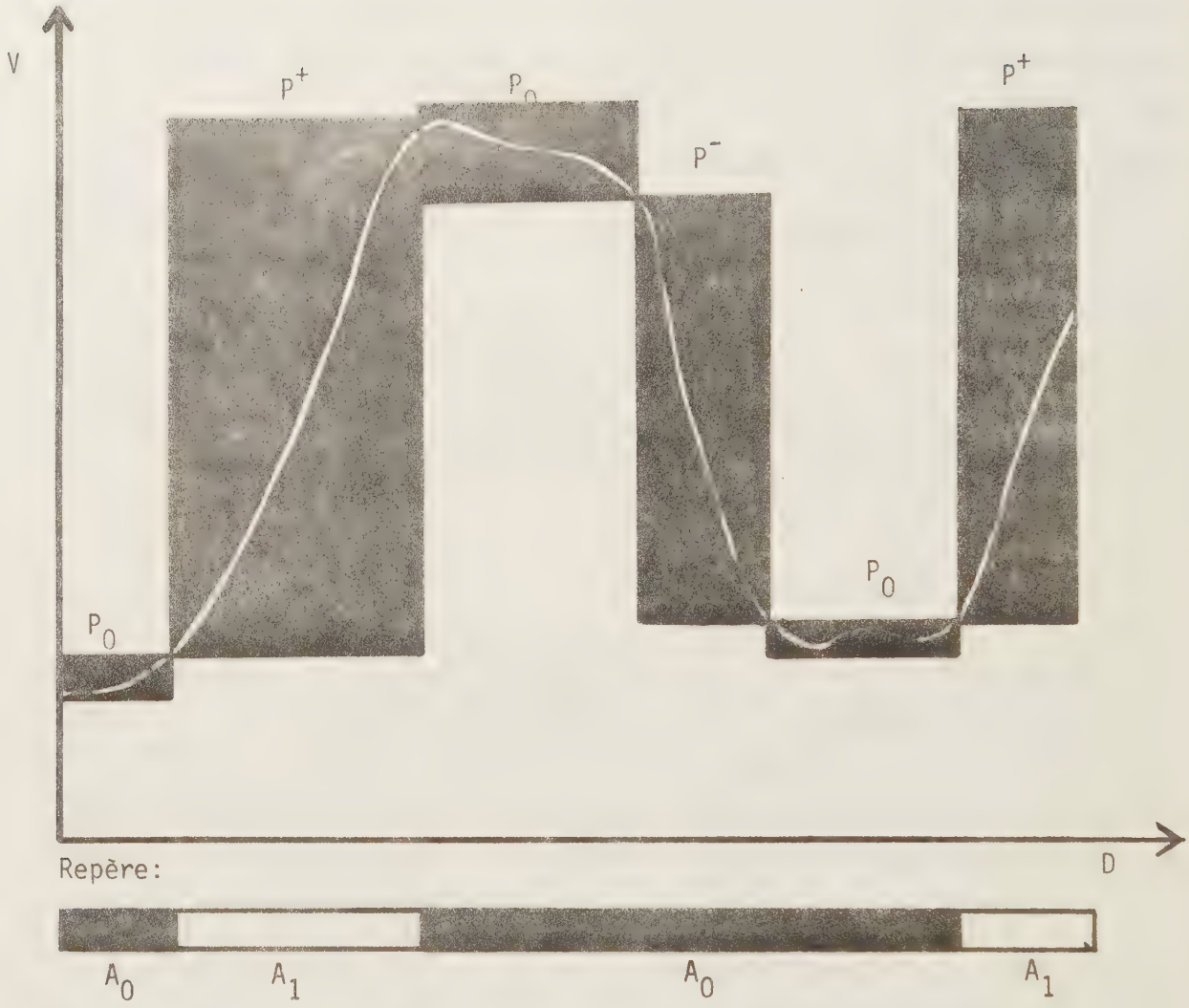
3) Soient $\delta = (t,m)$ et $T_x(\delta) = (t',m')$

alors il existe $a \in M(V,D)$ tel que :

$$m = a \frown m'$$

et $d(a) = t' - t = x \geq 0$.

Repère d'un signal continu



Partition de M(V,D):

P^+ : $\frac{dV}{dt} > +p_1$

P^- : $\frac{dV}{dt} < -p_2$

P_0 : $\frac{dV}{dt} \in [-p_2, +p_1]$

Alphabet:

A_0 : $P^- \cup P_0$

A_1 : P^+

II - LES REPERES

2.1. Définition

Dans la théorie des langages, les caractères n'ont pas de longueur propre et la longueur d'un mot est son nombre de caractères. Il n'en est pas de même pour les signaux physiques fonctions du temps, puisque le débit d'information - ie : le nombre de caractères par unité de temps - dépend de la durée des caractères. Il est donc nécessaire de spécifier cette durée, d'autant plus que, combinée à l'origine du signal, la suite des durées permet de comparer entre eux deux signaux, composante par composante. Cette suite et l'origine constituent un repère de signal dont nous donnons la définition suivante.

Soit $S = T \times M(V,D)$ un ensemble de signaux ;
Soit $P = (M_1,,M_i,,M_n)$ une partition finie de l'ensemble $M(V,D)$;
Soit $\mathcal{P}(P)$ l'ensemble des parties généré sur la partition P ;
Soit l'alphabet A un sous-ensemble fini de $\mathcal{P}(P)$ dont l'intersection avec la partition P n'est pas vide ;

Alors, une projection $P_A(\delta)$ du signal $\delta = (t,m)$ sur l'alphabet A est une suite* de composantes m_j de l'ensemble $M(V,D)$ telle que :

1) $m(\delta) = m_0 \frown m_1 \frown m_j \frown m_e$;

2) la longueur ℓ de la suite est finie si la durée D est finie,

3) la somme des durées des composantes est égale à D ,

4) toute composante m_j appartient à un et un seul élément de l'alphabet A;

5) la durée de toute composante m_j (à l'exception de la dernière si la durée D n'est pas finie) est bornée. (intuitivement, il ne peut en être autrement, car si la durée d'une composante n'est pas finie, l'information associée ne peut être acquise).

Soit $P_A(\delta)$ une projection d'un signal $\delta = (t,m)$ sur un alphabet A ;
Soit la suite d des durées des composantes m_j de la projection :

$$d = d(m_o),,d(m_j),,d(m_e) \quad ;$$

---

* une suite est dénombrable par définition [BOU]

Alors le doublet $r_A = (t,d)$ est le repère du signal $\delta = (t,m)$, relativement à l'alphabet A .

La longueur du repère est égale à la longueur de la projection $P_A(\delta)$.

Par analogie avec une composante de signal, nous appellerons composante d'un repère tout intervalle du type $[t + \sum_{i=0}^{n} d(m_i),\ t + \sum_{i=}^{n+1} d(m_i)[$

Le signal δ est alors caractérisé par le triplet :

$$\delta = (t,d,P_A)$$

Le signal δ admet en fait une infinité de repères ; il y a autant de repères que de façons de composer le signal par concaténation de composantes de l'ensemble $M(V,D)$.

Lorsque l'échelle de mesure du temps est définie directement à partir du signal, c'est à dire de l'alphabet A , tous ces repères sont indiscernables. Autrement dit, deux composantes adjacentes qui ne diffèrent que par leur durée sont indiscernables. La concaténation de caractères identiques de l'alphabet A a alors la propriété d'absorption :

$$a \frown a = a \quad .$$

Repères partiels

Un traitement de l'information n'est significatif que dans la mesure où les signaux - supports de l'information - ont un état défini aux moments du traitement. Par contre, leur état importe peu en dehors de ces moments, pourvu que le fonctionnement de la machine n'en soit pas perturbé.

La notion de repère partiel découle de ces remarques. Un signal se décompose alors temporellement en deux types de composantes :

- 1) les composantes définies , celles où le signal a une probabilité égale à 1 d'avoir une composante qui n'appartienne qu'à une seule classe de la partition de l'ensemble des composantes $M(V,D)$;

- 2) les composantes indéfinies pour lesquelles la probabilité précédente est strictement inférieure à 1. Ces composantes appartiennent cependant à une classe, la classe $M(V,D)$ de l'ensemble des parties $\mathcal{P}(M)$. Relativement à leurs éléments, les classes de cet ensemble $\mathcal{P}(M)$ n'ont malheureusement pas la propriété d'être disjointes ! La quantification de l'ensemble de base $M(V,D)$ ne correspond pas, ici, à sa partition.

2.2. Echelles de mesure

Nous venons de définir une quantification de l'ensemble des signaux, de telle façon que ses éléments soient dénombrables. Ceux-ci peuvent donc être un support d'information. Cependant, il apparaît, à travers la définition des repères de signaux, que la quantification est relative, car une infinité de repères peuvent être associés à un signal. Il importe alors de déterminer les paramètres dont dépend le dénombrement des classes possibles de signaux, c'est à dire la mesure de l'information véhiculée.

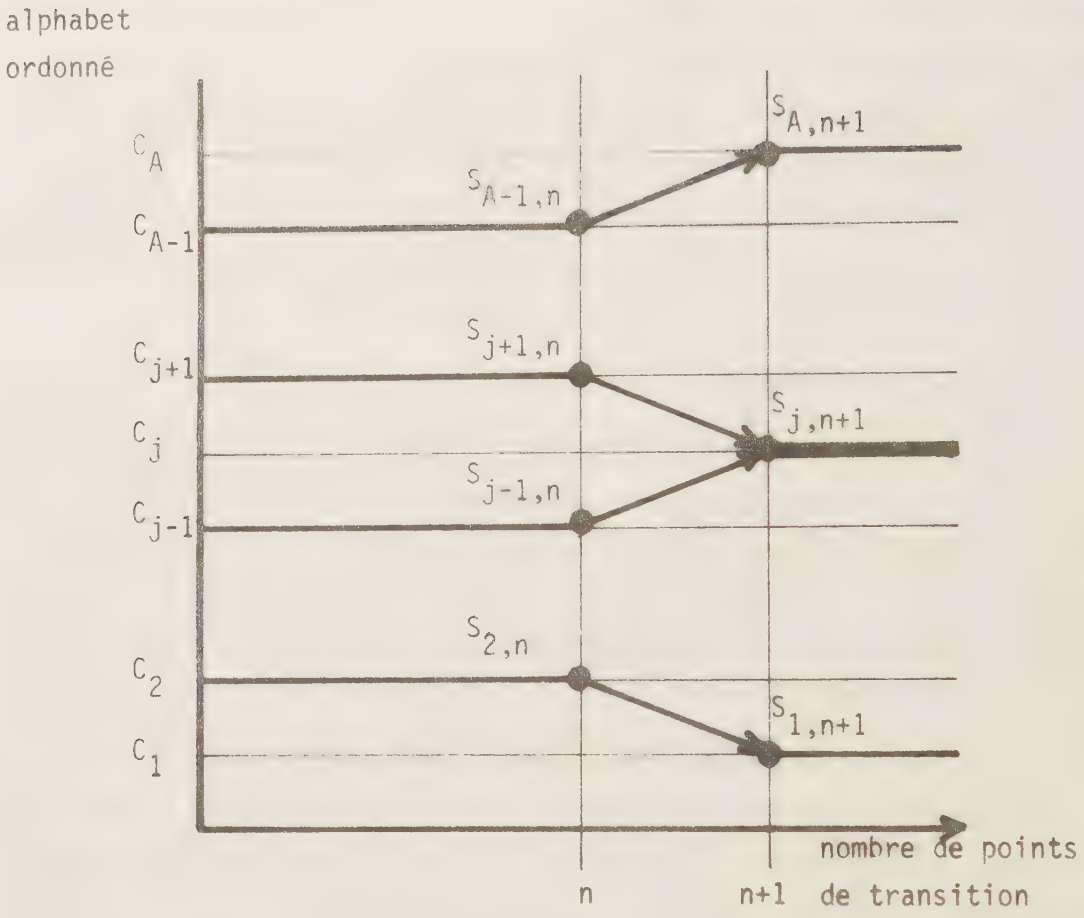
a) Mesure du temps

Dans le cas des signaux de l'ensemble $S = T \times M(V,D)$, chaque signal a une origine (t) et l'ensemble de ses composantes $M(V,D)$ est structuré en alphabet. Ce signal est donc muni d'au moins une échelle de mesure qui est la partition P de l'ensemble $M(V,D)$. Le dénombrement des signaux dépend alors du fait qu'une échelle de mesure est ou non associée à la grandeur temporelle T .

Lorsqu'il n'y a pas de référence (ie : échelle de mesure) associée au temps, le repérage d'un signal ne peut se faire que par rapport à ses points de transition, relativement à l'alphabet A . Nous disons que ce repérage est intrinsèque au signal. La propriété du repère résultant est que deux composantes adjacentes sont obligatoirement distinctes, puisque leur frontière commune est un point de transition. La longueur du repère est alors égale au nombre de transitions (y compris l'origine) ; le dénombrement des signaux consiste alors à dénombrer les enchaînements possibles de transitions.

Lorsqu'une référence temporelle est associée au signal, le nombre de composantes du repère est déterminé par cette échelle de mesure. La durée des composantes est alors égale à l'unité de mesure du temps. Le dénombrement des signaux nécessite, dans ce cas, l'identification des composantes, relativement à l'alphabet A. Le repère résultant ne peut être que partiel, car il n'y a aucune raison pour que les points de transition coïncident avec ceux de l'échelle de mesure. Celà implique qu'une composante du signal peut appartenir à plusieurs parties de la partition $P(M)$.

De la présence ou de l'absence d'une référence temporelle découlent donc deux dénombrements possibles des signaux. Si A est le nombre d'éléments de l'alphabet A, le nombre de repères possibles distincts associables à un signal est majoré dans chaque cas, par la valeur suivante, fonction du nombre N de composantes :



$S_{j,n}$: nombre de signaux à n points de transition qui se terminent par un caractère C_j

Dénombrement des signaux en fonction de leur nombre de points de transition, relativement à un alphabet ordonné

Repère intrinsèque : $n \leq (A-1)^N$,

Repère "externe" : $n \leq A^N$.

b) Alphabet ordonné

La majoration donnée pour un repère intrinsèque suppose que tous les caractères de l'alphabet A peuvent se suivre, pourvu qu'ils soient distincts. Cette valeur doit être minorée lorsque cet alphabet est ordonné et que le signal est continu (ce que nous avons supposé dès le départ).

Cette hypothèse est très fréquente pour les enregistrements magnétiques où :

1 - le signal est une tension induite fonction du temps : $v(t)$ et

2 - où le signal est autosynchronisé, c'est à dire lorsque la référence temporelle est reconstituée à partir des variations du signal (ex : mode d'enregistrement avec "retour à zéro").

Lorsque l'alphabet est strictement ordonné, un caractère ne peut avoir que deux successeurs distincts de lui-même ; exceptés le caractère le plus petit ou le plus grand qui n'ont qu'un successeur distinct possible. Le dénombrement des classes de signaux, en fonction du nombre de transitions peut être calculé de la manière suivante .

Pour un nombre donné de transitions n , on considère les sous ensembles de signaux qui se terminent par une composante dont le domaine de variation est inclus dans celui d'un même caractère de l'alphabet. Si A est le nombre de caractères de l'alphabet, il y a A sous-ensembles de ce type. On évalue alors le nombre de signaux caractérisés par $n + 1$ points de transition, en fonction du nombre de signaux qui possèdent n points de transition.

Soient $S_{1,n}; \ldots; S_{j,n}; \ldots; S_{A,n}$ les sous ensembles précédents, pour n points de transition ; ceux-ci sont liés aux mêmes sous-ensembles - pour $n + 1$ points de transition - par la relation suivante :

$$S_{j,n+1} = S_{j\ 1,n} + S_{j+1,n}$$

si j appartient à l'intervalle [2, A-1] ;

sinon $S_{1,n+1} = S_{2,n}$

et $S_{A,n+1} = S_{A+1,n}$.

Nous obtenons alors les relations suivantes :

$$S_{n+1} = 2.S_n - (S_{1,n} + S_{A,n})$$

$$S_1 = A(\forall_j S_{j,1} = 1)$$

Lorsque l'alphabet est binaire, le nombre de signaux distincts qui ont n points de transition est égal à 2 quel que soit n. Il en résulte qu'en logique binaire, un signal repéré intrinsèquement ne peut pas véhiculer de l'information (en fait, seulement un bit). Un tel signal ne peut servir que de référence temporelle ; ses points de transition délimitent alors les "unités" de temps. Nous en tirons la remarque très importante qu'en logique binaire : une référence temporelle ne peut pas véhiculer de l'information.

Il n'en est pas de même en logique ternaire car le nombre de signaux distincts qui ont n points de transition n'est pas un invariant ; il est égal à

$S_{2p} = 3.2^p$ si n est pair et égal à $2p$,

$S_{2p+1} = 4.2^p$ si n est impair et égal à $2p+1$.

Comme le repèrage temporel de ces signaux est intrinsèque, ceux-ci peuvent servir de référence temporelle. Il en résulte donc qu'en logique ternaire une référence de temps peut véhiculer de l'information

□●□●□●□●□●□

Pour conclure, nous voyons que la quantité d'information que peut véhiculer un signal - c'est à dire le nombre de classes distinctes de signaux - dépend du repèrage du signal et de la structure de l'alphabet.

III - ALGEBRE DES REPERES

1. Relation d'ordre

Nous avons construit un cadre - l'ensemble des signaux datés et repèrables - dans lequel la notion de mesure de l'information a un sens. Pour cela, nous sommes partis de l'ensemble des signaux $S = T \times M(V,D)$, équipotent à l'ensemble des réels. Nous avons construit ensuite une structure finie - l'alphabet A - sur l'ensemble des composantes $M(V,D)$, relativement à une partition P de cet ensemble $M(V,D)$. Enfin, nous avons introduit la notion de repère d'un signal. Un signal est alors caractérisé par sa date d'origine, une suite de caractères et la suite des durées de ces caractères.

Muni de cette structure, l'ensemble des signaux mesurables n'est cependant pas ordonné, alors qu'une structure d'ordre est nécessaire du point de vue logique, car elle permet d'induire les propriétés informatiques d'un signal de celles d'un autre.

Deux types d'inclusion (ie : d'ordre) sont définis :

1 - l'inclusion "complète" qui se rapporte aux points de transition des signaux ; elle permet de comparer des dates d'évènements ;

2 - l'inclusion "partielle" qui ne concerne que les repères partiels

a) Inclusion d'une composante :

Soient $r = (t,d)$ et $r' = (t',d')$ deux repères et soit c (resp. c') une composante du repère r (resp. r') . Nous dirons que la composante c est incluse dans la composante c'

1 - lorsque son origine temporelle absolue est supérieure ou égale à celle de c' ;

2 - lorsqu'elle "se termine" avant c' .

Une définition plus formelle est la suivante :

$$c \subseteq c' \Leftrightarrow \quad t + a(c) \geq t' + a(c')$$

$$t + a(c) + d(c) \leq t' + a'(c) + d(c')$$

où $a(c)$ est l'abscisse de c par rapport à son origine et $d(c)$ la durée de la composante.

r ⊂ r'

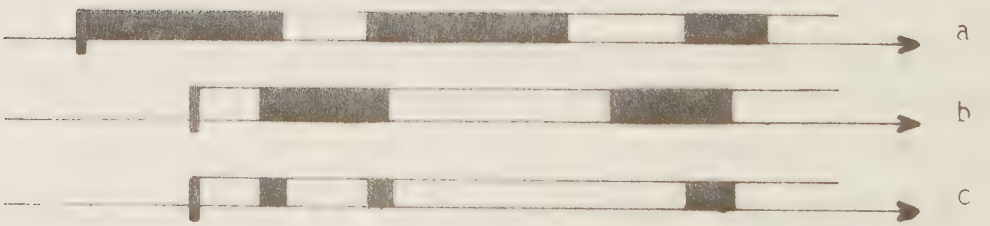
inclusion complète

r ⊂ r'

inclusion partielle

c = a ∪ b

union (repères partiels)



c = a ∩ b

intersection (repères partiels)

Cette relation est une relation d'ordre (réflexive, antisymétrique et transitive).

b) Inclusion complète

Un repère $\mathcal{r}$ est inclus dans un repère $\mathcal{r}'$ si chacune de ses composantes est incluse dans l'une du repère $\mathcal{r}'$. Cette définition entraîne la propriété suivante : A chaque composante du repère $\mathcal{r}'$ correspond une composante du repère $\mathcal{r}$ de même abscisse absolue $[t + a(c) = t' + a(c')]$.

c) Inclusion partielle

Soient $\mathcal{r} = (t,d)$ et $\mathcal{r}' = (t',d')$ deux repères construits sur les alphabets A et A'. Soient c et c' deux parties des alphabets A et A', telles que la partie c est strictement incluse dans l'alphabet A. Nous dirons que le repère $\mathcal{r}$ est inclus dans le repère $\mathcal{r}'$ si et seulement si chaque composante du repère $\mathcal{r}$, qui correspond à un caractère de c (cette composante est dite caractéristique), est incluse dans une composante du repère $\mathcal{r}'$, qui correspond à un caractère de c'.

La notion d'inclusion partielle est liée à celle de signal de commande. Un tel signal a la propriété de "concentrer" en lui toutes les contraintes temporelles, de durée et de position. Les relations temporelles entre les signaux doivent être telles que si les contraintes sont vérifiées par rapport au signal de commande, alors toutes les autres contraintes sont aussi vérifiées. Ces relations sont des relations d'inclusion temporelle qui doivent être vérifiées uniquement lorsque la commande est active (elle définit alors un évènement). L'état "inactif" de cette commande correspond au caractère neutre défini précédemment.

2. Lois de composition

Les lois d'union et d'intersection découlent directement de la relation d'inclusion. Elles ont la particularité de "modifier" les dates d'origine des repères.

a) Intersection

Soit R un ensemble de repères datés associé à l'ensemble des signaux $S = T \times M(V,D)$. Soient $\mathcal{r}_i = (t_i,d_i)_{i=1,n}$ n repères de R. Alors, l'intersection des repères $\mathcal{r}_{i=1,n}$ est le repère $\mathcal{r}(t,d)$ tel que :

1 - son origine t est la plus grande des origines t_i des repères $\mathcal{r}_{i=1,n}$

$$t(\mathcal{R}) = max(t_i)\ i = 1,n \quad ;$$

2 - le repère $\mathcal{R}(t,d)$ est le plus grand repère inclus à la fois dans tous les repères $\mathcal{R}_i(t_i , d_i)$:

a) $\forall\ i \mid i=1,n \ : \ \mathcal{R} \subseteq \mathcal{R}_i$,

b) $Si\ [\mathcal{R}' \mid \forall\ i \mid i = 1,n : \mathcal{R}' \subseteq \mathcal{R}_i)] \Leftrightarrow \mathcal{R}' \subseteq \mathcal{R}$

b) *Union*

La définition de l'union est "symétrique" à celle de l'intersection. Ainsi l'union des repères $\mathcal{R}_{i=1,n}$ est le repère $\mathcal{R}(t,d)$ tel que :

1 - son origine t est la plus petite des origines des repères $\mathcal{R}_{i=1,n}$: $t(\mathcal{R}) = min(t_i)\ i = 1,n$;

2 - chacun des repères $\mathcal{R}_i$ est inclus dans $\mathcal{R}(t,d)$ et l'union $\mathcal{R}(t,d)$ est le plus petit des repères qui inclut tous les repères $\mathcal{R}_i$:

a) $\forall\ i \mid i = 1,n \ : \ \mathcal{R}_i \subseteq \mathcal{R}$,

b) $[\mathcal{R}' \mid \forall\ i\ i = 1,n \ : \mathcal{R}_i \subseteq \mathcal{R}'] \Leftrightarrow \mathcal{R} \subseteq \mathcal{R}'$.

3. Treillis distributif

Les lois d'union et d'intersection précédentes sont telles que toute composante du repère union ou intersection de deux autres repères à les caractéristiques temporelles suivantes :

Intersection :

Soit X,Y,Z trois repères tels que $X \cdot Y = Z$.
Soient x,y,z trois composantes, appartenant respectivement aux repères X , Y et Z , telles que z est incluse dans x et dans y . Cette composante est telle que :

$$t_z = max(t_X , t_y)$$

$$T_z = min(T_X , T_y)$$

Avec t_z dénote la date du début de la composante et T_z la date de fin ($t_z = t_Z + a(z)$, $T_Z = t_z + d(z)$, où $a(z)$ dénote l'abscisse de la composante par rapport à son origine t_Z et $d(z)$ sa durée).

Union

Nous avons de même, si $X + Y = Z$, les relations suivantes :

$$t_z = min(t_x , t_y)$$
$$T_z = max(T_x , T_y) \quad .$$

Considérons alors les deux repères $V = (X + Y).Z$ et $W = X.Z + Y.Z$. Soient v et w deux composantes, de ces repères, qui correspondent aux mêmes composantes des repères X , Y et Z . Nous avons les relations :

$$t_v = max(min(t_x,t_y) , t_z)$$
$$t_w = min(max(t_x,t_z), max(t_y,t_z))$$

et
$$T_v = min(max(T_x,T_y) , T_z)$$
$$T_w = max(min(T_x,T_z), min(T_y,T_z))$$

Comme les fonctions réelles extrêmales sont distributives, nous obtenons les égalités :

$$t_v = t_w$$
$$T_v = T_w$$

La vérification de ces égalités par toutes les composantes des repères V et W permet d'en déduire l'égalité des repères - W et V :

$$(X + Y).Z = X . Z + Y . Z$$

Il en résulte que l'ensemble des repères, munis des lois d'union et d'intersection précédentes, a une structure de treillis distributif.

4. La mémorisation

Contrairement à l'union des repères qui s'applique principalement aux repères partiels, la mémorisation est une loi de composition qui ne s'applique qu'aux repères complets. Sa définition n'a de sens que dans l'ensemble des signaux c'est à dire lorsqu'on considère à la fois et le repère du signal et la valeur de ses composantes sur l'alphabet A .

Hypothèses :

Soit $S = T \times M(V,D)$ un ensemble de signaux ;

Soit P une partition de $M(V,D)$;

Soit A un alphabet construit sur la partition P ;

Soit R l'ensemble des repères de S, relativement à l'alphabet A ;

Soient s(resp.s') deux signaux de S , $p_A(s)$ et $r(s) = (t,d_A)$ une partition et le repère associé du signal s *(resp. s')* :

$$s = (t,d_A,p_A) \quad \text{et} \quad s' = (t',d'_A,p'_A)$$

La longueur du repère d'entrée r est supposée supérieure à 1, car la mémorisation n'a pas de sens pour des signaux constants .

Définition :
La mémorisation est une application f de S dans S telle que, si $s' = f(s)$:

1 - $t' = t + d(p_0)$ où p_0 la durée de la première composante de p_A ;

2 - $d'_A = (d_1,d_2 \ldots d_e)$ (alors que $d = d_0, d_1 \ldots d_e$)

3 - $p'_A = p_A = (p_0,p_1,\ldots p_e)$

Autrement dit, la mémorisation consiste à décaler d'un rang la "valeur" des composantes (ie : le caractère) et à décaler de même l'origine d'un rang.

Pseudo-complémentation.

La mémorisation est assimilable à une pseudo-complémentation, dans la mesure où :

1 - l'intersection d'un repère (r) et de son complémenté (r') est un repère neutre :

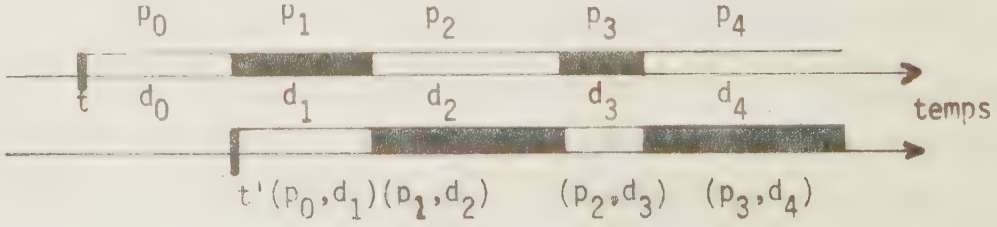
$$r.r' = (t,e) \ ,$$

$$t = max(t(r),t(r')) = t(r') \ ;$$

2 - l'union d'un repère (r) et de son complémenté (r') est un repère constant:

$$r + r' = (t(r),D) \quad .$$

La mémorisation n'est pas une complémentation ni une négation, car les éléments (t,e) et (t,D) ne sont pas uniques.



IV - LES FONCTIONS EXTREMALES

Nous venons de définir une algèbre temporelle des signaux, nous nous intéressons maintenant à l'algèbre "duale" des "alphabets".

1. Définition

Lorsque les fonctions combinatoires sont considérées comme des fonctions booléennes , le même alphabet est associé à l'état de tous les signaux, puisque les fonctions booléennes sont des lois de composition internes. De plus les fonctions booléennes sont définies de façon intrinsèque (par exemple, à l'aide d'une "table de vérité") et ne dépendent donc pas de leur contexte. Ces deux propriétés des fonctions booléennes se prêtent mal à l'étude de la logique de commutation.

D'une part, les alphabets associés à l'état des signaux n'ont pas toujours la même taille ; ainsi, l'alphabet associé à un "signal de données" peut être quelconque, alors qu'un signal de commande est en général binaire.

D'autre part, l'état (ie : effet) logique d'un signal de commande ne peut être défini à partir de l'élément qui le génère, mais à partir des éléments que le signal commande . Il en résulte que l'interprétation logique d'une fonction combinatoire ne peut être donnée à priori, car elle dépend du contexte lorsque cette fonction "manipule" des signaux de commande.

Comme la logique de commande s'exprime en termes d'état neutre et d'état actif, nous considérons les fonctions combinatoires - dans ce cas - comme des fonctions extrêmales des entrées. En effet, lorsqu'une commande est active, elle ne dépend pas, par définition, des autres entrées : elle correspond à un élément maximal . Par contre, une commande à l'état neutre n'interagit pas sur les autres entrées : elle correspond alors à un élément minimal , relativement à la hiérarchie des entrées.

Nous caractérisons alors les fonctions combinatoires, non pas comme des lois de composition interne, mais comme des applications f d'un produit d'alphabets $A_{i=1,n}$ dans un autre alphabet A :

$$A_1 \times \ldots \times A_i \times \ldots \times A_n \xrightarrow{f} A$$

Une fonction f est alors extrêmale sur un alphabet d'entrée A_i , si cet alphabet possède deux caractères m_i et M_i et il existe une application

h_i de $A_1 \times A_{i-1} \times A_{i+1} \times A_n$ dans A tels que pour tout uplet $\langle x_1, x_{i-1}, x_{n+1}, x_n \rangle$ de l'ensemble $A_1 \times A_{i-1} \times A_{i+1} \times A_n$:

caractère minimal : 1 - $f(x_1, x_{i-1}, m_i, x_{i+1}, x_n) = h_i(x_1, x_{i-1}, x_{i+1}, x_n)$

caractère maximal : 2 - $f(x_1, x_{i-1}, M_i, x_{i+1}, x_n) = a_{M_i}$

avec $a_{M_i} \in A$ et $h_i(x_1, x_{i-1}, x_{i+1}, x_n) \in A$.

Nous dirons qu'une fonction extrêmale est symétrique pour deux de ses entrées i et j si :

1 - $h_i(m_i) = h_h(m_i)$,

2 - $a_{M_i} = a_{M_j}$.

Les fonctions booléennes d'union et d'intersection sont alors des cas particuliers de fonctions extrêmales symétriques, lorsque tous les alphabets sont confondus ($\forall i = 1,n$: $A_i = A$) .

Ainsi, lorsque m est le plus petit élément, noté 0 , et M le plus grand, noté 1, alors f est l'union :

$$\forall x \in A : f(0,x) = x$$
$$f(1,x) = 1$$

Avec les conventions inverses, la fonction f est l'intersection :

$$\forall x \in A : f(1,x) = x$$
$$f(0,x) = 0$$

2. Porte logique

Il existe un autre type important de fonctions extrêmales, celui des portes logiques. Une fonction p de ce type est une application, de l'ensemble produit de deux alphabets D et C dans un autre S , telle que si l'alphabet C comporte deux éléments (c,C) et qu'il existe une bijection t de D dans une partie de S, alors :

$\forall d \mid d \in D$, $\exists m \in S$: 1) $p(C,d) = m$ et m est état neutre relativement aux successeurs de la porte,

2) $p(c,d) = t(d)$,

3) $S = m_u\ t(D)$.

La notion d'extrema (m,M) n'a pas de sens pour l'alphabet d'entrée D , relativement à la fonction de porte. Par contre, cette notion a un sens, relativement aux fonctions successeurs de la porte logique.

La fonction de porte peut être symétrique ou dissymétrique.

Les fonctions booléennes d'union et d'intersection peuvent être considérées comme des portes - 1) si les conventions suivantes sont adoptées pour une fonction de A^2 dans A, avec $A = \{0,1\}$:

union : $c = 0$, $C = 1$, $\delta = 1$,

intersection : $c = 1$, $C = 0$, $\delta = 0$;

- 2) si la condition suivante est vérifiée :

- pour l'union, le successeur de la porte doit être une fonction d'intersection,

- pour l'intersection, le successeur de la porte doit être une fonction d'union.

Ces contraintes d'emploi des fonctions booléennes, en vue d'exprimer la logique de commutation, nous ont poussé au choix d'un autre formalisme, pour étudier les éléments de mémorisation.

3. Initialisation logique

La fonction de porte logique permet d'"inhiber" un signal de commande. Cette fonction a un "double" , celle qui permet d'"activer" un signal de commande, quelque soit l'état de celui-ci. Une telle fonction sert à initialiser un système séquentiel.

Une fonction d'initialisation logique est une application (i) de l'ensemble produit de deux alphabets D (données) et C (commande) dans un autre S(sortie), telle que si l'alphabet C comporte deux éléments extrêmaux (c,C) et qu'il existe une bijection t de D dans une partie de S , alors :

$\forall d | d \in D$, $\exists M \in S$: 1) $i(C,d) = M$ et M est un état actif, relativement aux successeurs de la fonction,

2) $i(c,d) = t(d)$,

3) $S = M \cup t(D)$.

Cette fonction peut être symétrique ou non.

Les fonctions d'initialisation sont rarement employées comme telles, car l'effet de la commande n'est pas inversé (la sortie est active quand la commande l'est). Il est en effet plus simple de réaliser physiquement des fonctions inverseur. Lorsqu'elle est utilisée, cette fonction apparait sous la forme d'un "forçage" des sorties, en général par mise à la masse à travers un élément qui a une impédance ou bien très élevée ou bien très faible.

Par contre les fonctions symétriques de ce type ont une application fondamentale : la convergence (éléments à collecteur ouvert de la logique TTL) .

E_2 \ E_1	m	M
m	m	M
M	M	M

Initialisation symétrique

C :	c	C
D :	D	M

Initialisation dissymétrique

4. Jeu de fonctions extrêmales.

Nous nous limitons à un alphabet binaire {neutre, actif} , noté {*m*,M}, pour tous les signaux de commande. A l'inverse des fonctions booléennes, l'état d'un signal n'est pas interprété par rapport à la source, mais par rapport aux récepteurs du signal.

Cela suppose que l'effet d'un signal est le même pour tous les récepteurs. Lorsqu'il n'en est pas ainsi, nous sommes obligés d'introduire, dans le schéma logique, une fonction de complémentation qui transforme un état neutre en état actif et vice-versa. Cette obligation n'est pas incohérente, car elle traduit le fait qu'un même état peut avoir deux effets opposés.

En plus de la complémentation, nous ne retenons comme fonctions de base, que les deux fonctions extrêmales suivantes :

- la porte symétrique (graphisme) ,
- la porte dissymétrique (graphisme).

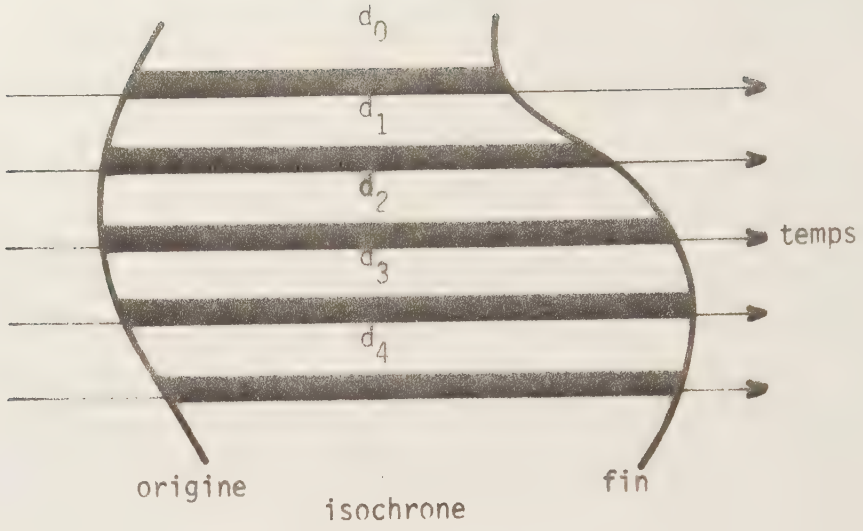
E_2 \ E_1	m	M
m	M	m
M	m	m

porte symétrique

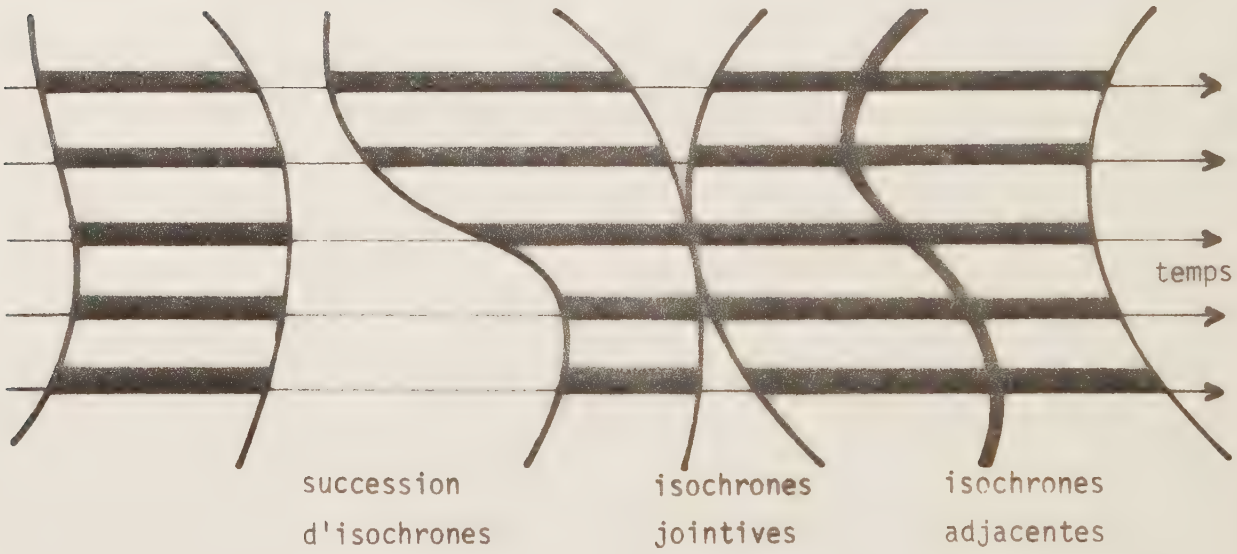
C:	m	M
D:	D	m

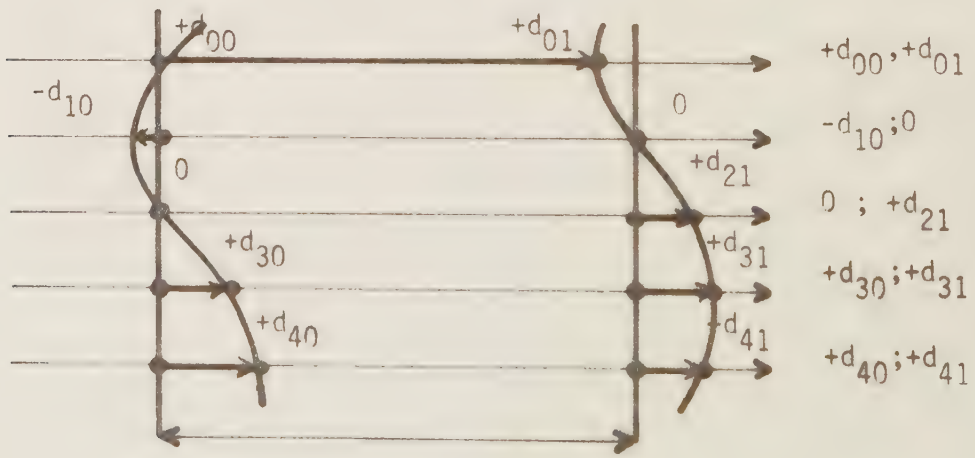
porte dissymétrique

Ce jeu de fonctions est suffisant pour étudier le fonctionnement des éléments de mémorisation et est indépendant de la technologie.



LES ISOCHRONES





système de coordonnées d'une isochrone

V - LES ISOCHRONES

Dans les parties précédentes, nous avons défini l'état d'un signal comme un mot sur un certain monoïde et relativement à un repère. Nous introduisons maintenant la notion d'isochrone, afin d'étendre le concept de mot à l'ensemble produit des ensembles de signaux $S = T \times M(V,D)$.

5.1. Définition

Une composante datée $c_i = (t_i , d_i , m_i)$ d'un signal daté(s) de l'ensemble $S = T \times M(V,D)$, relativement à un monoïde A^* défini sur $M(V,D)$, est telle que si $s = (t,m)$:

1) $\exists a , b \in A^* : m = a^\frown m_i ^\frown b$

2) $t_i = t + d(a)$

3) $d_i = d(m_i)$

Une composante datée vide est toute composante dont la durée est nulle ou bien dont la composante m est l'élément vide du monoïde A^* .

Soient $S_{i=1,n}$ n ensembles de signaux et soient $A^* i = 1,n$ des monoïdes associés. Soit X un n-uplet de signaux de l'ensemble produit $S_1 x S_i x S_n$.

Une isochrone I , sur l'ensemble produit $S_1 \times S_i \times S_n$ est un n-uplet de composantes datées des signaux X_i :

$$I = (C_1 , C_i , C_n)$$

avec $C_i|i=1,n| = (t_i,d_i,m_i)$, $m_i \in A_i^*$.

L'état de l'isochrone est le n-uplet (m_1,m_i,m_n) , c'est à dire un élément de l'ensemble produit des monoïdes A_i^* construits sur les alphabets A_i :

$$A_i^* = A_i^0 \cup A_i^1 \cup \ldots \cup A_i^p \cup \ldots$$

Une isochrone non repérée est une isochrone dont seul l'état est défini.

L'origine de l'isochrone est le n-uplet des origines des composantes datées: (t_1,t_i,t_n). De même, l'épaisseur de l'isochrone est le n-uplet des durées des composantes : (d_1,d_i,d_n) . La fin d'une isochrone estle n-uplet des dates de fin de ses composantes : $F = (t_1+d_1 , t_i+d_i , t_n+d_u)$. L'origine et la fin d'une isochrone en constituent les frontières inférieure et supérieure.

Les coordonnées d'une frontière $F = (t_1,t_i,t_n)$, relativement à une date t, est le n-uplet de coefficients réels k_i tel que :

$$\forall i = 1,n \ : \ t_i = t + k_i$$

Deux isochrones a et b sont jointives si la frontière supérieure de l'isochrone "gauche" a et la frontière inférieure de l'isochrone "droite" b , notée respectivement (T_1,T_i,T_n) et (t_1,t_i,t_n) , vérifient les contraintes suivantes :

1 - $\forall i = 1,n : T_i \leq t_i$

2 - $\exists i \in [1,n] : T_i = t_i$

Deux isochrones sont adjacentes si elles sont jointives en tout point : $\forall i = 1,n : T_i = t_i$. Relativement à une même date, la frontière supérieure de l'isochrone a et la frontière inférieure de b ont alors les mêmes coordonnées.

Deux frontières a et b se succèdent lorsque toutes les dates de a sont inférieures à celles de b , composante par composante :

Soit $a = (T_1,T_i , T_n)$

$b = (t_1,t_i , t_n)$

alors b succède à a (noté a,b) si :

$$\forall i = 1,n : T_i \leqslant t_i$$

Il est bien évident que la frontière supérieure d'une isochrone succède à sa frontière inférieure, car la durée des composantes ne peut être négative.

Nous étendrons les lois de composition d'union, d'intersection et de complémentation de l'algèbre des repères à un ensemble d'isochrones, de la manière suivante : - les composantes non vides d'une isochrone sont considérées comme les composantes définies d'un repère partiel , alors que les composantes vides correspondent à des composantes indéfinies. Une isochrone correspond ainsi à une "tranche" dans le temps d'un produit de repère.

Pour caractériser une isochrone en termes d'origine et de durée d'un état, les coordonnées des frontières sont exprimées en fonction de deux paramètres réels t et d tels que :

- 1) les coordonnées de la frontière inférieure sont exprimées par rapport à la date t,

- 2) les coordonnées de la frontière supérieure sont exprimées,
- ou bien par rapport à la date t pour les composantes d'établissement, causes de l'origine de l'état ;

- ou bien par rapport à la date $t + d$ pour les composantes de maintien, causes de la durée de l'état.

L'ensemble des isochrones qui ont les mêmes coordonnées relativement aux paramètres (t,d) constituent une famille d'isochrones.

La concaténation $A \frown B$ de deux familles d'isochrones A et B est la famille d'isochrones $a \frown b$ jointives des deux familles :

$$a \frown b \in A \frown B \Leftrightarrow a \text{ et } b \text{ sont jointives à gauche de } b .$$

Lorsque deux isochrones a et b de $A \frown B$ sont adjacentes , il en est de mêmes pour tous les couples $a \frown b$ de la famille $A \frown B$, puisque toutes les isochrones de la famille A (ou B) ont les mêmes coordonnées. Nous dirons dans ce cas que la concaténation $A \frown B$ est adjacente.

5.2. Relation de transfert

La notion d'isochrone permet de caractériser tout état d'un système séquentiel. Il reste alors à définir la relation qui lie les états de départ (cause) et d'arrivée (effet) d'un changement d'état ; c'est pourquoi nous introduisons la relation de transfert. Cette relation est une relation d'ordre strict , c'est à dire non réflexive ; elle est par contre antisymétrique et transitive.

Soit $I(S)$ l'ensemble des isochrones associé à l'ensemble produit des signaux S. Cet ensemble est muni de la structure d'ordre strict suivante, notée $\rightarrow$:

1- antisymétrie : $(a,b \in I \mid a \rightarrow b \text{ et } b \rightarrow a) \Rightarrow a = b$;

2- transitivité : $(a,b,c \in \mid a \rightarrow b \text{ et } b \rightarrow c) \Rightarrow a \rightarrow c$;

La relation $a \rightarrow b$ se lit "transfert de a dans b" . Cette structure d'ordre, ainsi que la structure inverse, correspond au principe de causalité : b est l'effet de la cause a .

5.3. Etats

Nous définissons maintenant un état d'un système séquentiel à l'aide de couples d'isochrones : (cause (C) , effet (E)). Pour tous ces couples, l'isochrone - "effet" est identique et complète , c'est à dire qu'elle inclut tous les effets .

Ainsi, lorsque $a \rightarrow b$

et $a \rightarrow c$

$$\exists e,x,y,x',y' \in I : e = x^{\frown}by = x'^{\frown}cy'$$

et $a \rightarrow e$.

La multiplicité des couples (causes, effet E) correspond à l'éventualité de causes multiples et indépendantes.

L'union de deux isochrones-causes est aussi une isochrone-cause.

a) Etats initiaux

Un système est toujours supposé initialisable. Autrement dit, on peut lui associer des états initiaux du type suivant :

$\exists i , \delta \in I$ tels que : $i \rightarrow \delta$ et :

$\forall j \mid j = 1,n$: soit $i_j = \emptyset$ et $\delta_j \neq \emptyset$ (δ_j correspond à un signal interne ou de sortie)

soit $i_j \neq \emptyset$ et $\delta_j = \emptyset$ (signal d'entrée)

et $\nexists a,b \in I : a \rightarrow b$ avec $b_j = i_j$

b) Etats propres

A l'exception des systèmes combinatoires, les systèmes séquentiels ont des états propres . Ces états correspondent à la propriété de mémorisation. Ils sont caractérisés par au moins deux couples d'isochrones ; l'un correspond à l'établissement et l'autre au maintien. Ce dernier couple définit une règle récursive , dépendante du contexte ; il traduit le fait que le maintien d'un état suppose son existence préalable.

$\exists e, m, s \in I$ tels que :

établissement : $e \rightarrow s$

maintien : $m \,.\, s \rightarrow s$

La double relation précédente sera écrite, par la suite , sous la forme :

$$s \leftarrow e \,/\, m$$

5.4. Synchronisme

a) Horloge

Les signaux générés par une horloge non interruptible (ie : du type oscillateur) sont caractérisés par des isochrones du type suivant :

Si $h \in I$ et $h \rightarrow h$

et $x \rightarrow h \Rightarrow x = h$

alors $h^* \in I$ où h^* est l'itéré de h par concaténation :

$h^* = h \,\bar{}\, h \,\bar{}\, \ldots \,\bar{}\, h \,\bar{}\, \ldots .$

Soit l'horloge H l'ensemble des isochrones de ce type.

b) Machine synchrone

L'ensemble I associé à une telle machine possède une partie H dont les éléments sont les plus petites isochrones de I :

$$\forall i \in I \mid i \notin H \,,\; \exists x \in I \,,\; h \in H : x \,.\, h \rightarrow i$$

Cela implique que les signaux d'horloge sont les seuls signaux d'entrée ! Tous les autres signaux sont alors repérés par rapport à eux. Une machine purement synchrone conduit à un système fermé !

c) Machine asynchrone

Pour une telle machine, les isochrones d'entrée sont les plus petits éléments de I . Nous revenons en fait à la définition d'une machine synchrone, à l'exception de la relation de périodicité des signaux d'entrée.

Nous avons donné ces deux définitions pour en montrer l'inutilité, car nous avons postulé* qu'il existait un système de repérage commun à tous les signaux d'une machine, y compris les signaux d'entrée .

Autrement dit, le concept de machine est indissociable de celui de repérage. Les termes synchrone et asynchrone ne spécifient en fait, pour une machine, que l'origine du repérage :

- interne à la machine (synchrone) ou bien
- externe à la machine (asynchrone).

5.5. Système séquentiel

Un système séquentiel est défini par un ensemble d'états qui constitue son "alphabet" et par des règles d'enchaînement de ces états .

Par concaténation des isochrones-effets, il doit alors être possible de générer complétement l'ensemble des signaux défini comme sortie d'état du système.

Celà implique que ces isochrones-effets sont adjacentes . Par contre, cette contrainte ne se retrouve pas sur les causes : les repères résultant de la concaténation des causes peuvent être partiels. Il s'agit en effet de déterminer les causes minimales d'un état. Les composantes indéfinies des causes sont telles que leur état n'influe pas sur celui du système : la notion de hiérarchie des signaux en découle. Par exemple, un signal d'initialisation est actif, quelque soit l'état des autres signaux d'entrée.

Le fonctionnement logique d'un système séquentiel peut donc être défini comme un langage, celui des états ; on peut alors poser le problème d'"acceptation d'une suite d'états par le système séquentiel".

Les règles d'acceptation sont simples et sont au nombre de deux :

Soit E l'ensemble des états du système .

(*) Cette hypothèse est contenue dans la notion d'isochrone.

Si a et b sont deux états du système, alors la suite a,b est aussi un état du système, si et seulement si :

$$a , b \in E : a = (e , e) \quad , \quad b = (e',e')$$

$$e \rightarrow e \quad , \qquad e' \rightarrow e'$$

1 - La concaténation $e \frown e'$ est <u>adjacente</u> ;

2 - e' <u>succède à</u> e : e , e' .

Lorsque les <u>coordonnées</u> des isochrones sont exprimées en fonction des temps de retard introduits par les éléments logiques, ces clauses d'acceptation permettent de calculer les contraintes temporelles et de détecter les <u>aléas</u>.

Nous distinguerons deux sortes de contraintes :

1 - celles sur la durée des composantes : les contraintes <u>dynamiques</u> ,
2 - celles sur la valeur des caractéristiques temporelles des éléments logiques : les <u>contraintes statiques</u>.

A cause de la dispersion des valeurs caractéristiques, il n'est pas toujours possible d'assurer ces contraintes statiques. Elles sont, par là, plus importantes que les contraintes dynamiques qui peuvent être toujours respectées (lorsqu'il s'agit d'une minoration des durées).

NOTE 1

Alphabet Morse

Cet exemple illustre le problème de repèrage que soulève le choix d'un alphabet.

L'alphabet Morse comprend deux caractères :

- le point,
- le trait,

et deux réparateurs :

- le "silence" court,
- le silence long (qui sert à délimiter les lettres). Ces quatres caractères ne sont pas indépendants.

Leur classification relève d'une double partition :

1 - partition binaire du domaine de variation :

- impulsion "sonore" et "silence",

2 - partition binaire de la durée des composantes :

- brève et - longue.

Deux composantes du signal ne sont discernables que si leur domaine de variation est différent. Ainsi deux "silences" brefs sont indiscernables d'un silence long.

Si l'origine du signal est synchronisée sur une composante "sonore", un alphabet de quatre caractères indépendants peut être défini ; il diffère du précédent :

1 - point ⁀ silence bref, (durée minimale)
2 - point ⁀ silence long,
3 - trait ⁀ silence bref,
4 - trait ⁀ silence long. (durée maximale)

Cette partition exclut la fonction de séparateur. Un des quatre caractères peut être affecté à cette fonction ; mais alors, cette affectation dépend uniquement du langage défini sur cet alphabet.

Une propriété fondamentale distingue ces deux alphabets : le second est indépendant du langage véhiculé, car chaque caractère est discernable. Par

contre, les quatre composantes (point, trait, silence bref ou long) ne forment pas un alphabet indépendant. Une composante ne peut être comptée pour un caractère que relativement à la syntaxe d'un langage (qui est alors lié à son alphabet). La syntaxe doit alors exclure la contiguïté de deux composantes qui ont le même domaine de variation.

□●□●□●□●□

NOTE 2

Algèbre de Boole et logique des évènements

Dans une algèbre de Boole, les lois de Morgan permettent de définir une des deux lois d'union ou d'intersection à partir de l'autre et de la loi de complémentation. Relativement à des signaux, ces relations sont vérifiées lorsque l'on considère leurs états. Par contre, il n'en est plus ainsi lorsque ces signaux sont considérés comme des suites d'évènements. La complémentation des états ne permet plus de passer de l'une à l'autre loi d'union ou d'intersection.

Ces deux lois peuvent alors s'exprimer en termes de début et de fin d'évènements. Ainsi, l'union d'évènements E_i est un événement U qui débute dès que un des évènements E_i apparait et dure tant que tous les évènements E_i ne sont pas finis. De même, l'intersection d'évènements E_i est un des évènement I qui ne débute pas tant que tous les évènements E_i ne sont pas simultanément présents et se termine dès que un des évènements disparait.

Il résulte de la relation "tant que tous" de l'union que la fin d'un évènement de durée finie peut être "retardée" d'une durée quelconque jusqu'à la fin d'un évènement retardateur. Par contre, la même relation "tant que tous" de l'intersection n'implique pas que le début d'un évènement de durée finie - donc temporaire - puisse être retardé d'une durée quelconque. L'évènement résultant n'a lieu en effet que si tous les évènements sont présents y compris les évènements temporaires. La symétrie entre l'union et l'intersection ne peut alors être obtenue qu'en transformant un évènement en un état : l'opération de complémentation est alors temporelle : il s'agit de la mémorisation.

Algèbre de Boole

Les trois lois de l'algèbre des repères sont telles qu'il existe une date à partir de laquelle l'algèbre résultante confère aux repères les propriétés

d'une algèbre de Boole . Certaines conditions doivent alors être vérifiées :

1 - les repères sont complets, relativement à une partition du type précédent $P = \{e, D\}$;

2 - la longueur des repères est supérieure à 1 ;

3 - deux signaux, constants à partir de la date d'observation mais dont la "valeur" est complémentaire (ie : e et D) doivent être supposés comme étant chacun le repère "mémorisé" de l'autre .

4 - les repères constants $r(e)$ et $r(D)$ sont les éléments minimal et maximal de l'algèbre "de Boole".

5 - les lois de composition sont notées de la façon suivante

• intersection , + union , ' mémorisation .

Dans ces conditions, soient a , b , c de tels repères et soit la date d'observation $t = max(a', b', c, (a.b)'$ et $(a+c)')$. A partir de cette date, les relations suivantes sont vérifiées :

$a \cdot D = a$	$a + e = a$
$a \cdot a' = e$	$a + a' = D$
$a \cdot b = b \cdot a$	$a + b = b + a$
$a \cdot (b+c) = (a \cdot b)+(b \cdot c)$	$a +(b \cdot c) = (a + b) \cdot (a + c)$
$(a \cdot b)' = a' + b'$	$(a + b)' = a' \cdot b'$ (lois de Morgan).

Nous dirons que cette algèbre vérifie les relations d'une algèbre Boole en "régime permanent" . Ces relations ne sont pas vérifiées à l'initialisation (que nous appellerons le "régime transitoire").

CHAPITRE III

Ce chapitre est consacré aux systèmes séquentiels à " mémoire ". Ceux-ci possèdent plusieurs états stables dont le repérage est lié à celui des entrées et des sorties. La synthèse et l'analyse du " noyau " de ces systèmes - la bascule RS - sont effectuées à partir d'une décomposition du système en éléments combinatoires. Les systèmes à mémoire sont ensuite classés d'après les propriétés de leur mécanisme d'échantillonnage ainsi que de leur système de repèrage des sorties d'état.

APPLICATION DU MODELE AUX SYSTEMES A MEMOIRE:

ETUDE DES CELLULES DE MEMOIRE STATIQUES

I - La bascule RS

A - Principe de l'étude des bascules

B - Echantillonnage

C - Transparence

LES CELLULES DE MEMOIRE

Notations

Afin d'étudier les états d'une cellule de mémoire, en termes de repères, de fonctions extrêmales, de temps de retard et d'isochrones, nous avons adopté les notations suivantes.

1) Etat

- a) Un signal, son repère associé ou bien les composantes du repère sont notés à l'aide de la même lettre. La distinction provient du contexte dans lequel la lettre est employée.

- b) Une lettre minuscule caractérise un repère partiel (ou une composante) minimal ; par contre,

- c) une lettre majuscule caractérise un repère partiel (ou une composante) maximal.

2) Temps

La date d'origine d'une composante est caractérisée par la lettre minuscule t, alors que sa date de fin correspond à la majuscule T. Ces deux caractères sont indicés par le nom de la composante.

La durée d'une composante est représentée par la lettre indicée d. Nous avons alors la relation suivante :

$$T_{\cdot} - t_{\cdot} = d_{\cdot}$$

Les coordonnées des isochrones sont exprimées sous deux formes différentes en fonction des besoins.

- 1) couples (t_i, d_i) exprimés par rapport aux paramètres (t,d) de l'isochrone. Les valeurs par rapport à un référentiel absolu sont alors :

$$(t + t_i , d + d_i)$$

- 2) couples (t_i , T_i) exprimés par rapport aux dates des frontières de l'isochrone (t , T). Les valeurs par rapport à un référentiel absolu sont alors:

$$t + t_i , T + T_i \text{ ; durée: } (T - t) + (T_i - t_i).$$

Les temps de retard sont supposés identiques à la montée et à la descente; ils sont notés par la lettre θ.

I - LA FONCTION "IDEALE" DE MEMORISATION

La mémorisation est l'opération fondamentale qui permet de s'affranchir du temps : elle fait coïncider temporellement des évènements qui sont générés séquentiellement !

De là découle son nom. Cependant cette appellation nous semble impropre dans la mesure où cette opération rejette en fait la notion d'état permanent. L'opération consiste en effet à décaler d'une position dans le temps une suite d'évènements. Autrement dit, un évènement n'est mémorisé que jusqu'à l'apparition de celui qui lui succède, lequel, à son tour est mémorisé. Cette opération de base n'est donc qu'une mémorisation temporaire.

Cependant, la "mémorisation" est distincte de la fonction de retard. A l'opposé d'un évènement retardé, la durée d'un évènement mémorisé est indépendante de la durée de cet évènement. Elle dépend uniquement de la durée - supposée quelconque - qui sépare celui-ci de l'évènement successeur. Bien que distinctes, ces deux opérations sont cependant liées. Il est ainsi possible de définir l'opération de mémorisation comme une composition de retards, au prix d'une contrainte : la durée (donc la fréquence) des évènements mémorisés est alors liée à celle des "temps de retard". Par contre, la réciproque n'est pas vraie. Cette dissymétrie résulte des "principes" qui sont sous-jacents à la notion de mémorisation.

Le décalage temporel (ie : la mémorisation) d'une suite d'évènements ne peut en effet être réalisé que de deux façons.
1) retarder chaque évènement jusqu'à l'apparition du suivant et celà sans modifier sa durée ;
2) étendre la durée d'un évènement, jusqu'à l'apparition du suivant.

La première méthode est incompatible avec le principe selon lequel "l'effet suit la cause". La durée du retard (l'effet) ne dépendrait plus de l'évènement mais du suivant, lequel serait alors la cause d'un effet qui le précède ! La

notion de durée d'un évènement "mémorisé" est ainsi relative à une suite d' évènements. C'est ce qui empêche de passer de la notion de mémorisation à celle de retard.

La réciproque est par contre possible, elle est le fondement sur lequel sont construites les mémoires à "circuits intégrés". Le principe de base est le suivant: l'itération périodique d'un "évènement" dont la durée est égale à la période constitue un phénomène constant. Un tel phénomène est physiquement réalisable: il suffit de combiner un signal à son "retardé" d'un temps inférieur à la durée de l'"état" observé du signal, état qui constitue l'évènement.

Le mécanisme décrit ne "mémorise" que la première occurrence de l'évènement, son état est ensuite permanent et indépendant des occurrences suivantes. Cependant, s'il n'y a qu'un type d'évènement, on ne peut logiquement espérer plus d'un mécanisme de mémorisation. Car la signification d'une telle fonction est uniquement liée à l'observation des évènements de "sortie" relativement à ceux d'"entrée". Comme l'état permanent de sortie correspond à la présence d'un évènement d'entrée, la correspondance est vérifiée pour chaque évènement d'entrée. La fonction de décalage temporel n'est effectivement réalisée que lorsque l'"alphabet" associé aux signaux comporte plusieurs caractères.

Dans ce cas, le repère du signal source doit être égal au plus grand repère inclus dans tous les repères de ce signal. Car la fonction de décalage temporel est telle qu'il existe une bijection entre la suite d'évènements d'entrée et la suite mémorisée. Les repères partiels sur lesquels un des évènements d'entrée n'apparaît pas doivent être exclus.

De plus, le décalage temporel ne peut être réalisé à l'aide de fonctions combinatoires que si la combinaison du signal d'entrée et du signal retardé est une fonction extrêmale des "caractères" d'entrée par rapport aux caractères de sortie. Il faut en effet qu'un caractère d'entrée puisse "effacer" le caractère précédemment mémorisé, afin de le remplacer.

La définition d'une telle fonction de mémorisation est alors la suivante

Soit E un signal et S = f(E, S) son image par la mémorisation f,

Soient R_E et R_S les repères associés, relativement aux alphabets A_E et A_S qui sont :

$$A_E = \{(E_1, \ldots E_i \ldots E_n), e\}$$

$$A_S = \{S_1, \ldots S_i \ldots S_n\}$$

Alors, la fonction de mémorisation a les propriétés suivantes :

1) Retard : $t(R_S) = \theta + t(R_E)$ (dates d'origine)

2) Relation entre le retard et la durée des évènements :

$$\forall i \ | i = 1,n | : \ d(E_i) \geqslant \theta$$

3) fonction extrêmale

$$\forall i \quad : f(e,S_i) = S_i$$

$$\left| \begin{array}{l} \forall i = 1,n \\ \forall j = 1,n \end{array} \right. : \quad f(E_i,S_j) = S_i$$

Initialisation : il découle, de cette définition de la mémorisation, que le signal S est "indéfini" avant la date d'origine $t(R_E) + \theta$

II - EQUATION DE MEMORISATION

Nous nous limitons à la logique binaire et nous supposons que la cellule de mémoire a une sortie d'états.

Chaque état a une origine et une durée. A chacune de ces caractéristiques nous associons un repère partiel d'entrée ; le repère d'établissement (E) caractérise l'origine de l'état et le repère de maintien (m) est associé à sa durée. Les composantes définies de ces repères partiels déterminent les contraintes minimales d'établissement et de maintien de l'état.

Le repère de sortie (s) est un repère complet combinaison de ces deux repères partiels. Le repère s doit être complet, puisque le but de cette étude est de déterminer les contraintes d'entrée qui garantissent un état stable en sortie, état partout défini à partir de l'origine.

Les composantes définies du repère de maintien sont minimales (m) car l'état résultant du repère de sortie est dû à la coïncidence (ie : intersection) de ces composantes de maintien avec ses propres composantes définies. Celà implique que les composantes définies du repère s sont donc elles aussi minimales.

Par contre, les composantes définies du repère d'établissement E sont maximales (M), puisqu'elles permettent la mise en état, que les clauses de maintien soient vérifiées ou non.

Le repère de sortie s est donc l'union du repère d'entrée E et du repère (noté I) - intersection du repère s lui-même et du repère de maintien (m). Les composantes définies du repère intermédiaire I doivent être elles aussi maximales, afin que le repère -"union" s soit complet.

Le repère de sortie a alors pour expression :

$$S \leftarrow E + s.m$$

Cette équation de repères est l'équation de mémorisation. Elle permet de calculer les contraintes temporelles, lorsque les fonctions extrêmales sont combinées à des fonctions de retard.

L'équation précédente peut être vérifiée à l'aide d'une boucle de deux portes logiques symétriques. D'après les équivalences que nous avons données dans le chapitre précédent, les couples de fonctions booléennes qui vérifient cette équation sont les suivants :

- 1) couple OU(+) et ET (.) ;
- 2) couple "NAND","NAND";
- 3) couple "NOR" ,"NOR".

III- ISOCHRONE DE MEMORISATION

Nous calculons maintenant les coordonnées de l'isochrone qui vérifie l'équation précédente, en fonction des temps de retard introduits par les fonctions extrêmales. Nous choisissons comme paramètres de l'isochrone, la date d'origine t_E du repère d'établissement E et la durée d_m de la composante définie du repère de maintien m. Nous notons θ_E (retard à l'établissement) le retard dû à l'union E + I ; nous notons θ_m (retard du maintien) le retard dû à l'intersection I ← S.m

3.1. Initialisation

D'après la définition des lois d'union et d'intersection, appliquée à l'équation de mémorisation, nous obtenons la relation temporelle suivante :

$$t_s = \theta_E + \min (t_E, \theta_m + \max (t_s, t_m))$$

Nous supposons que l'effet (ie : l'état s) suit la cause (ie. l'établissement E) ; nous avons donc à l'initialisation :

$$t_E < t_s$$

D'autre part, le repère de maintien doit coïncider avec le repère s et ne doit pas "inhiber" l'état de mémorisation, donc :

$$t_m \leqslant t_s \qquad (3)$$

L'origine du repère de sortie est alors, d'après la relation 1 ;

$$\boxed{t_s = t_E + \theta_E} \qquad (4)$$

L'origine du repère de maintien doit donc être inférieure à :

$$\boxed{t_m \leqslant t_E + \theta_E}$$

3.2. Mémorisation

La mémorisation implique que le repère I ait une date d'origine inférieure ou égale à la date de disparition de la composante définie du repère d'établissement E, afin que l'union des repères I et E soit un repère complet :

$$t_I \leqslant T_E \quad (= t_E + d_E)$$

Puisque $t_I = \theta_m + \max(t_s, t_m)$ et d'après les relations 3 et 4, nous déduisons la contrainte temporelle sur la durée minimale de la composante d'établissement E :

$$\boxed{d_E \geqslant \theta_E + \theta_m} \qquad (7)$$

3.3. Durée

La durée de la composante définie du repère de sortie s peut être exprimée en fonction de celle de la composante de maintien, si l'on suppose que l'état de mémorisation est atteint (ce qui suppose la vérification de la relation 7). Le repère de sortie n'est alors complet que depuis l'origine et jusqu'à la fin de l'état de mémorisation.

Si T_s, T_E, T_m dénotent les dates de fin des composantes définies, nous avons les relations suivantes :

$$T_s = t_s + d_s$$
$$T_m = t_m + d_m$$
$$T_s = \theta_E + \max(T_E, \theta_m + \min(T_s, T_m))$$

La fin de la composante définie s n'est due qu'à la disparition de la cause de maintien. Comme l'effet suit la cause,

$$T_m < T_s$$

D'autre part, la date T_E de la composante d'établissement correspond aux contraintes minimales, donc :

$$T_E = t_E + \theta_E + \theta_m$$

Enfin, puisque $T_m = t_m + d_m \leqslant t_E + \theta_E + d_m$ et que

$$T_s = t_s + d_s = t_E + \theta_E + d_s$$

nous obtenons finalement l'expression de la durée de la composante définie s, en fonction de celle de la composante de maintien :

$$d_s \leqslant d_m + \theta_E + \theta_m$$

3.4. Repère intermédiaire I

L'origine de ce repère est dueau signal de sortie s (relation 3) et sa durée est due à celle de la composante de maintien m. Les caractéristiques de la composante définie du repère I sont alors les suivantes :

$$\text{origine} : t_I = t_s + \theta_m = t_E + \theta_E + \theta_m$$
$$\text{Durée} : d_I = d_m + \theta_m - \theta_m = d_m$$

3.5. Coordonnées

Les relations temporelles précédentes nous permettent de déterminer complètement l'isochrone minimale de mémorisation :

$$s \leftarrow E/m$$

Etat :			
	E	composante définie maximale	causes
	m	composante définie minimale	
	I = s.m	composante définie maximale	effets
	s	composante définie minimale	

Temps :	
E	$\underline{t_E},\ d_E \geqslant \theta_E + \theta_m$
m	$\leqslant \underline{t_E} + \theta_E,\ \underline{d_m}$
I	$\underline{t_E} + \theta_E + \theta_m,\ \underline{d_m}$
s	$\underline{t_E} + \theta_E,\ d_s \leqslant \underline{d_m} + \theta_E + \theta_m$

La date $\underline{t_E}$ et la durée $\underline{d_m}$ caractérisent la famille des isochrones de mémorisation. Relativement à ces paramètres les coordonnées de l'isochrone sont alors :

Frontière inférieure : (par rapport à t_E)

E	=	0
m		θ_E
I		$\theta_E + \theta_m$
s		θ_E

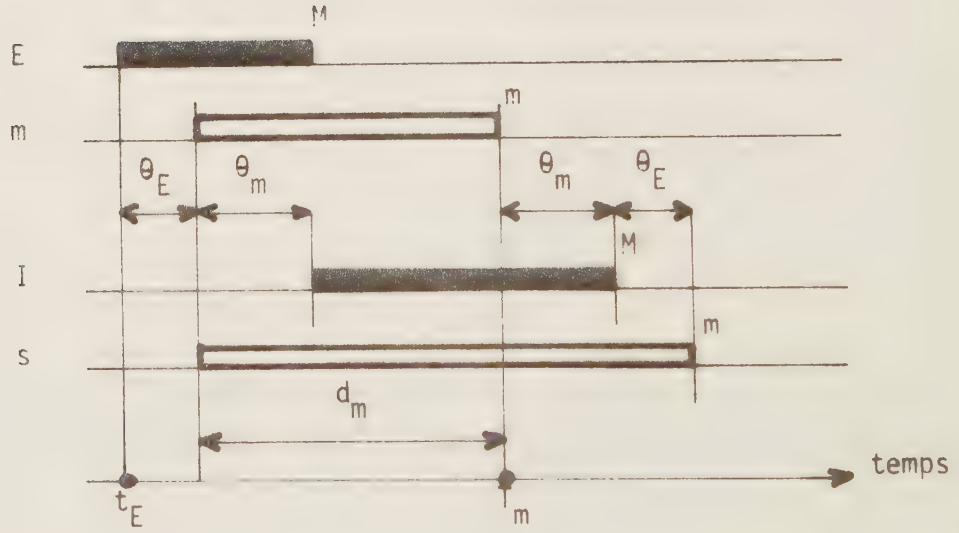
t_E

frontière supérieure :
(par rapport à t_E et d_m)

$$\left|\begin{matrix} E \\ m \\ I \\ s \end{matrix}\right| = \left|\begin{matrix} \theta_E + \theta_m \\ — \\ — \\ — \end{matrix}\right|_{t_E} ; \left|\begin{matrix} — \\ 0 \\ 0 \\ \theta_E + \theta_m \end{matrix}\right|_{d_m}$$

Nous regroupons ces coordonnées sous les deux formes suivantes :

$$\left|\begin{matrix} E \\ m \\ I \\ s \end{matrix}\right| = \left|\begin{matrix} (0,\ \theta_E + \theta_m) ; — \\ \theta_E \ ; \ 0 \\ \theta_E + \theta_m \ ; \ 0 \\ \theta_E \ ; \ \theta_E + \theta_m \end{matrix}\right|_{t_E \ ; \ d_m} = \left|\begin{matrix} (0,\ \theta_E + \theta_m) ; — \\ \theta_E \ ; \ 0 \\ \theta_E + \theta_m \ ; \ \theta_m \\ \theta_E \ ; \ \theta_E + \theta_m \end{matrix}\right|_{t_E ; \ T_m}$$



isochrone de mémorisation

IV- AUTOMATE SEQUENTIEL

4.1. Symétrie

D'après les équations précédentes, il apparaît que les deux repères d'établissement et de maintien ne sont pas mutuellement exclusifs, par rapport à une échelle de temps absolue. Ils doivent donc être associés à deux signaux physiquement distincts.

D'autre part, nous n'avons étudié que les conditions qui permettent d'obtenir un seul état de mémorisation, alors que la cellule de mémorisation doit être bistable, afin de "mémoriser un bit d'information".

Le second état de mémorisation peut être obtenu simplement par symétrie : il suffit de permuter les rôles des deux signaux d'établissement et de maintien. Soient R et S ces deux signaux d'entrée. Deux repères sont alors associés à chacun d'eux : l'un d'établissement et l'autre de maintien. Ces deux repères sont dans ce cas mutuellement exclusifs dans le temps et le repère résultant associé au signal en est l'union.

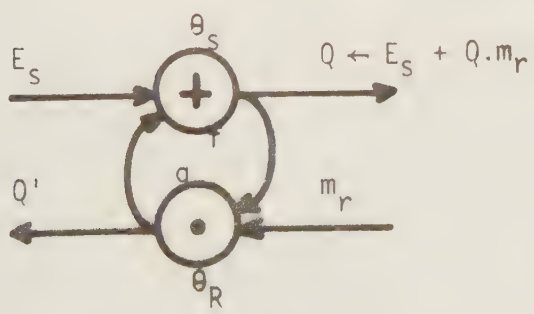
$$\text{Repère } R \leftarrow E_R + m_r$$

$$\text{Repère } S \leftarrow E_s + m_s$$

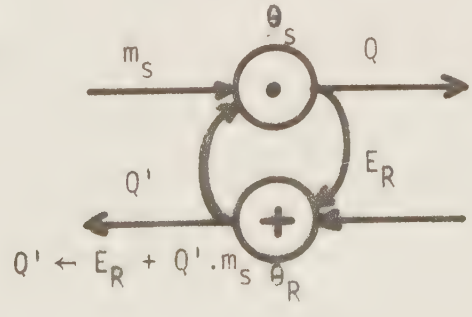
La cellule de mémorisation comporte alors :

- deux signaux d'entrée R et S ;
- deux fonctions extrémales f et g ;
- deux signaux de sortie Q et Q' .

Cette structure est symétrique. La mise dans chacun des deux états est alors caractériséepar les deux schémas symétriques suivants :



état " SET " (S)



état " RESET " (R)

4.2. Graphe des états

Cette cellule de mémorisation possède deux états propres (de mémorisation) , mais elle n'est pas pour celà un système séquentiel. Il faut encore vérifier que la succession de deux états propres quelconque est un état propre de la cellule.

Par définition, la concaténation d'un état propre avec lui-même est un état propre de la cellule. En effet, tant que la cause de maintien existe, l'effet - ie : l'état propre - existe. La cause d'établissement peut disparaître, réapparaître : une fois l'état propre atteint, la cause d'établissement n'interagit plus sur la cellule - tant que la cause de maintien existe.

Nous avons donné, dans le chapitre précédent, la condition nécessaire et suffisante pour que deux états propres distincts se succèdent :
- 1) la concaténation des deux isochrones - "effets" doit être adjacente et
- 2) les isochrones de cause doivent se succéder.

Comme la structure de la cellule est symétrique, il suffit de vérifier une seule des deux successions possibles : (S,R) ou (R,S). La vérification est immédiate. Il suffit de calculer les coordonnées de la frontière supérieure du premier état (nous prendrons S), par rapport à la date de fin d'une composante - "effet"(Q) et de les comparer à celles de la frontière inférieure du second état (R), mesurées par rapport à la date de début de la même composante - effet (Q). Comme les composantes - effets doivent être jointives, nous devons obtenir les mêmes coordonnées pour les effets, puisque la date de référence est la même. De plus nous devons vérifier que les dates des causes se suivent ou bien sont identiques.

suite d'états (S,R) : les isochrones S et R de mémorisation ont pour coordonnées.

Isochrone S :

$$\begin{vmatrix} S \\ R \\ Q' \\ Q \end{vmatrix} = \begin{vmatrix} (0,\ \theta_S + \theta_R) \\ \theta \\ \theta_S + \theta_R \\ \theta_S \end{vmatrix}_{t_S} ; \begin{vmatrix} - \\ 0 \\ 0 \\ \theta_S + \theta_R \end{vmatrix}_{d_R}$$

Isochrone R :

$$\begin{vmatrix} S \\ R \\ Q' \\ Q \end{vmatrix} = \begin{vmatrix} \theta_R \\ (0,\ \theta_R + \theta_S) \\ \theta_R \\ \theta_R + \theta_S \end{vmatrix} ; \begin{vmatrix} 0 \\ - \\ \theta_R + \theta_S \\ 0 \end{vmatrix}$$

t_R $\qquad$ d_S

De ces coordonnées, nous allons déduire les coordonnées de la frontière supérieure de S, mesurées par rapport à la fin de la composante définie Q_S ; ainsi que celles de la frontière inférieure de R, mesurées par rapport au début de la composante Q_R.

Pour l'isochrone de gauche S, il suffit d'ajouter les coordonnées sur t_S à celles de d_R et de retrancher la valeur obtenue pour la composante Q de référence. Pour la composante de droite R, il suffit de retrancher aux coordonnées sur l'origine t_R, la valeur de la composante Q. Nous obtenons ainsi :

Frontière supérieure de S

$$\begin{vmatrix} S \\ R \\ Q' \\ Q \end{vmatrix} = \begin{vmatrix} -d_S - \theta_S \\ -(\theta_R + \theta_S) \\ -\theta_S \\ 0 \end{vmatrix}$$

$q\downarrow$

Frontière Inférieure de R

$$\begin{vmatrix} S \\ R \\ Q' \\ Q \end{vmatrix} = \begin{vmatrix} -\theta_S \\ -(\theta_R + \theta_S) \\ -\theta_S \\ 0 \end{vmatrix}$$

$Q\uparrow$

Nous voyons que les dates pour les "effets" Q et Q' sont identiques. Les dates de la composante - cause R sont aussi identiques. Par contre, nous obtenons une contrainte dynamique pour le signal S, puisque les dates $T_Q(S)$ et $t_Q(R)$ doivent se suivre:

$$d_S \geqslant 0$$

Nous avons déjà supposé que cette contrainte était vérifiée. Il en résulte que les états S et R se suivent bien. Par symétrie, il en est de même de la suite R, S. La cellule de mémoire qui vient d'être décrite est bien un système séquentiel qui a deux états propres R et S. Cette cellule est en fait une bascule RS.

V - CAS LIMITE DE FONCTIONNEMENT

Nous venons de voir que la cellule de mémoire précédente a deux états propres et qu'une suite de ces deux états est "acceptée" par la cellule. Du point de vue de l'utilisateur, nous devons encore définir la cadence limite à laquelle des états propres distincts peuvent se suivre ; autrement dit, nous devons déterminer la durée minimale de chaque suite d'états distincts qui peut être itérée.

La durée de l'isochrone, associée à une telle suite d'états, doit être identique pour chaque signal - effet, puisque les composantes définies de ce type de signal sont adjacentes. La période d'une suite d'états est alors la somme des durées minimales des composantes définies d'un même signal effet.

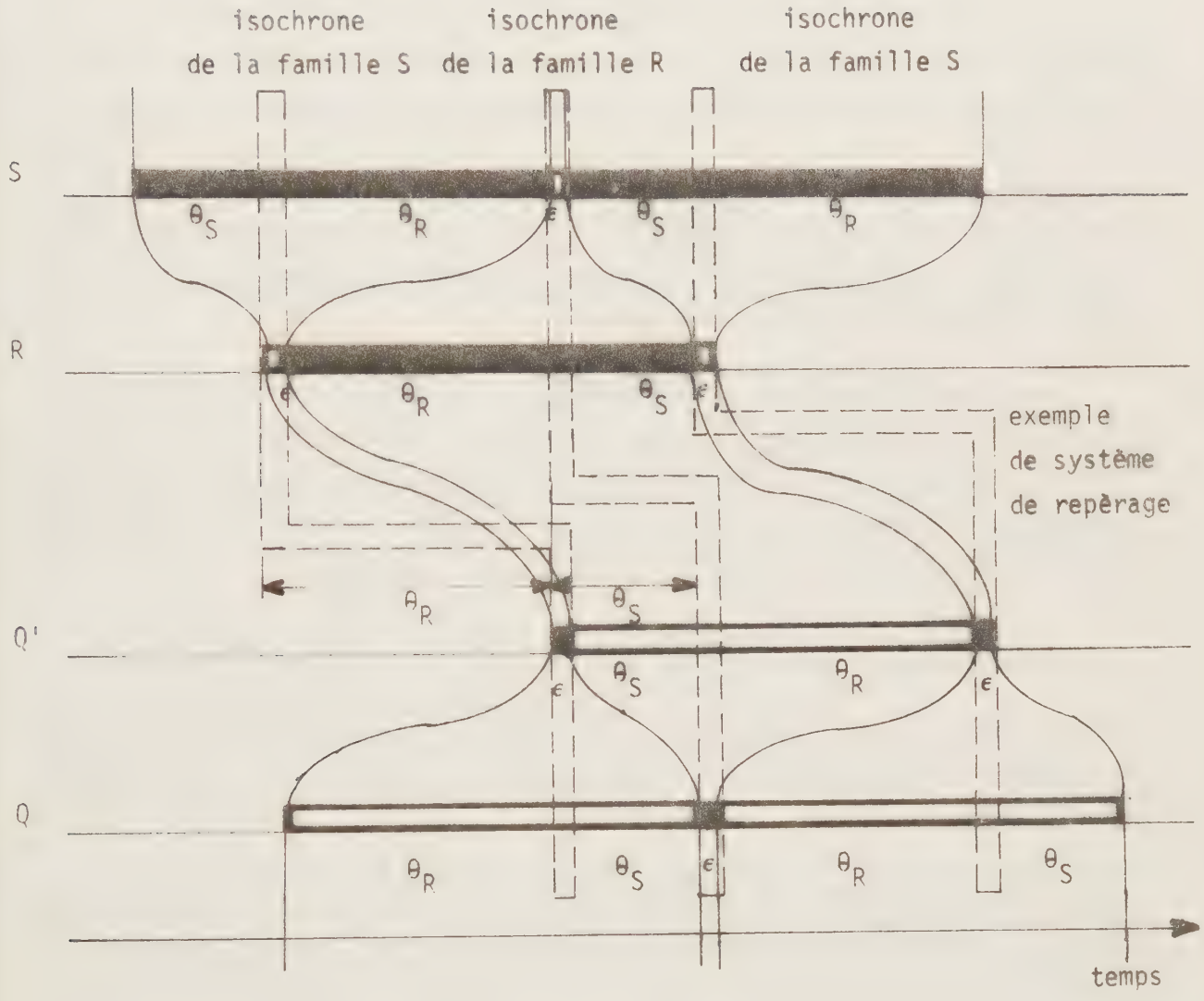
Nous avons choisi de caractériser n'importe quelle isochrone, en particulier celle qui caractérise un état propre, par deux paramètres : - la date t et la durée d de l'isochrone. Pour répondre à la question précédente, il suffit de faire tendre vers sa limite inférieure la durée des isochrones associées aux états propres. Les coordonnées d'une isochrone déterminent alors la position relative et la durée minimale de chaque composante. Le système de coordonnées ainsi obtenu regroupe en fait la somme des contraintes temporelles associées à un état propre.

Pour la cellule de mémorisation, nous avons déterminé la limite inférieure de la durée de chaque état ; cette limite correspond à celle de la durée de la composante de maintien : d_s ou $d_r \geqslant 0$.

La durée minimale d'une suite d'états distincts (R,S) ou (S,R) est alors égale à :

$$d(R,S) = d(S,R) \geqslant \theta_R + \theta_S$$

L'isochrone minimale d'une telle suite d'états a alors la forme suivante .



SUITE D'ETATS PROPRES

(cas limite de fonctionnement)

VI - REPERAGE

A la suite des calculs précédents, nous pouvons affirmer que la cellule fonctionne comme un élément de mémorisation, lorsque les contraintes temporelles que nous avons déterminées sont respectées. Cependant, le raisonnement que nous avons poursuivi ne nous dit pas comment le fonctionnement de la cellule peut être interprété comme une opération de mémorisation. Ce genre d'interprétation nécessite la définition d'un système de repérage.

Jusqu'alors nous avons supposé que nous étions synchronisés sur les signaux et que nous pouvions discerner les différentes composantes. Cette aptitude mentale à la synchronisation nécessite - dans la réalité - un point de repère physique, afin de se synchroniser.

Comme nous l'avons dit précédemment, ce point de repère ne peut être qu'un point de transition de l'état d'un signal, par rapport à une référence. D'autre part, ce point de repère ne peut logiquement appartenir qu'à un signal d'entrée, car l'état - donc les points de transition - des sorties ne peut être prévisible puisque celles-ci "véhiculent" l'information.

Le système de repérage le plus élémentaire est alors constitué
1) d'un point de transition d'un signal d'entrée,
2) d'une date "d'échantillonnage" de la sortie, repérée par rapport au point de transition. Cette date est en fait un temps de retard, puisque l'effet suit la cause.

De plus, un système de repérage n'a de sens que par rapport aux sorties d'état du système. Autrement dit, il faut définir par rapport à quel signal l'état du système est observé. Dans le cas de la cellule étudiée trois sorties d'état sont possibles :

- le signal Q,
- le signal Q',
- les deux signaux Q et Q'.

Lorsque la sortie d'état est effectuée à l'aide de plusieurs signaux ceux-ci peuvent être observés simultanément ou bien à des dates différentes, pourvu que l'état des signaux à ces dates d'observation détermine uniquement un et un seul état du système.

Sur le schéma du paragraphe précédent, nous avons dessiné en pointillé un tel système de repérage. Les points de repère délimitent les composantes minimales de maintien. La sortie d'état est supposée constituée des deux signaux Q et Q'. Cette sortie est observable avec un temps de retard θ_R pour l'état S et un temps de retard θ_S pour l'état R.

Pour le cas limite étudié, la composante de maintien (p. ex :R) est incluse temporellement dans la composante d'établissement (p. ex : S), de même, la durée de la composante intermédiaire (p. ex : Q') est incluse dans l'autre composante de sortie (p.ex : Q). De plus, l'état des composantes S ou Q' est complémentaire de celui de la composante R ou Q. Le système de repérage indiqué permet donc de définir de façon biunivoque l'état de la cellule de mémorisation.

Cependant, ce système de repérage n'a guère d'intérêt, puisque le repérage est effectué à partir des composantes de maintien. Comme la durée de ces composantes définit celle de l'état de mémorisation, la "lecture" des sorties - par rapport à ces composantes - n'apporte alors aucune information.

Le seul système de repérage logique est en fait le suivant,
-1) le repère d'entrée doit caractériser de façon biunivoque chaque composante d'établissement d'un état propre,
-2) le repère de sortie doit être le complémentaire du repère d'entrée, de la façon décrite dans la première partie.

Comme l'entrée est composée de deux signaux à effet symétrique, le repère d'entrée ne peut être que l'union des repères d'établissement associés à chaque signal d'entrée. De plus, pour que l'interprétation des composantes du repère d'entrée soit unique, les composantes d'établissement R et S doivent être mutuellement exclusives dans le temps.

Le cas limite de fonctionnement physique ne peut donc être logiquement atteint.

Avec cette nouvelle contrainte temporelle, les isochrones de mémorisation ont alors pour coordonnées :

$$\begin{vmatrix} E \\ m \\ S.m \\ S \end{vmatrix} = \begin{vmatrix} 0, \ (\theta_E + \theta_m) \leqslant d_R \\ 0 \\ \theta_E + \theta_m \\ \theta_E \end{vmatrix} \quad ; \quad \begin{vmatrix} - \\ 0 \\ 0 \\ \theta_m + \theta_E \end{vmatrix}$$

t_E $\qquad$ $dR \geqslant \theta_m$

La durée minimale d'une suite (R,S) ou (S,R) devient alors égale à :

$$d(R,S) \geqslant 2\,(\theta_S + \theta_R)$$

Nous retrouvons des résultats de la théorie des machines séquentielles qui peuvent être déduits de la définition logique des bascules "RS". Cependant, l'interprétation des résultats est entièrement différente. En effet, nos calculs ont montré que cette bascule "RS" fonctionne même dans le cas indéterminé théorique (c'est à dire le cas où les deux composantes d'établissement coïncident temporellement en un point). La mutuelle exclusion temporelles des composantes d'établissement ne résulte donc pas d'un problème de fonctionnement, il résulte uniquement d'un problème de repérage, c'est à dire d'interprétation logique du fonctionnement physique.

VII - REGIME OSCILLATOIRE

Pour que l'état de mémorisation soit atteint, nous avons supposé que le repère de sortie devait être complet. Nous en avons déduit la durée minimale de la composante d'établissement. Lorsque cette durée est inférieure à la durée limite, l'état atteint est encore stable, car la stabilité n'est due qu'à la composante de maintien. L'état obtenu est périodique et la période est la somme des temps de retard :

$$T = \theta_E + \theta_m$$

Puisque le repère de sortie n'est pas complet sur la période T : la bascule RS oscille. Ce régime de fonctionnement doit être évité à tout prix.

Malheureusement, de telles oscillations ne peuvent - logiquement - être inhibées dans une boucle de mémorisation. Il faut alors introduire des para-

mètres supplémentaires dans le modèle logique. Le modèle de J. Suchard devient alors nécessaire : pour que des oscillations, ne puissent se produire, il faut que les composantes dont la durée est inférieure à la période $\theta_E + \theta_m$ soient filtrées. La solution est simple : il faut que le temps de montée des signaux soit supérieur aux temps de retard introduits par les éléments logiques.

Il n'y a donc aucun intérêt à ce que les signaux aient des "fronts de montée ou descente très raides" : cette propriété est en effet un inconvénient.

Deuxième partie ETUDE DES BASCULES

A - PRINCIPE DE L'ETUDE

Avant d'étudier les propriétés des bascules, nous exposerons le principe de la méthode employée. Son fondement est le suivant :
- chaque élément est prédéfini comme un automate d'états,
- la synthèse de sa structure - vue comme un ensemble de fonctions extrêmales - est faite en termes d'établissement et de maintien d'un état,
- l'analyse de ses propriétés a pour but de déterminer les contraintes temporelles - évaluées à partir des caractéristiques des fonctions extrêmales qui garantissent la fonction logique d'automate d'états.

L'étude des propriétés d'une bascule est pour celà décomposée en celle de séquences d'états élémentaires de la bascule. Une propriété est alors définie par un ensemble de séquences d'états qui doivent être "acceptées" par la bascule. D'autre part, chaque état élémentaire est spécifié par l'état de tous les signaux caractéristiques d'entrée et de sortie.

La vérification d'une propriété consiste alors :
1) A vérifier que les états élémentaires appartiennent à "l'alphabet" de la bascule. Cette appartenance implique qu'aucune relation lie l'état des signaux non caractéristiques . Il en résulte que les composantes de ces signaux ne peuvent apparaître que dans des expressions temporelles d'union et qu'elles doivent être éliminées lors de l'analyse.
2) A vérifier que chaque état d'une séquence succède à son prédécesseur.
3) A vérifier que les séquences d'états peuvent se succéder entre elles.

Les vérifications précédentes impliquent le calcul des coordonnées de l'isochrone caractéristique d'un état. L'énoncé du problème doit alors absolument définir par rapport à quels paramètres temporels ces coordonnées sont calculées. Il est en effet impossible d'obtenir un système cohérent de relations temporelles sans point de repère.

1 . Contraintes temporelles

Comme dans l'étude de la cellule de mémoire, le calcul des coordonnées

détermine les contraintes temporelles que doivent vérifier certains signaux. Ces contraintes sont évaluées par rapport aux paramètres de l'isochrone, de telle façon que les coordonnées minimales de toute composante ne soient fonction que des paramètres de l'isochrone.

Ainsi, l'intersection de deux composantes, dont les coordonnées de l'une (c) sont les paramètres, introduit les contraintes suivantes pour les coordonnées de l'autre composante (X):

$$\text{isochrone } (t_c, T_c) : R \leftarrow x.c,$$
$$\text{alors } t_x \leqslant t_c,$$
$$T_x \geqslant T_c.$$

Par contre, l'union de deux composantes n'introduit pas de contrainte temporelle qui lie les entrées entre elles. Les seules contraintes qui sont introduites sont alors des contraintes "de cause à effet".

Par exemple, si 1) $u \leftarrow X + Y$ et si 2) l'état X caractérise l'isochrone et si 3) les paramètres de l'isochrone sont les coordonnées de la composante u, alors la relation d'union introduit les contraintes suivantes : $t_X \leqslant t_u - \theta$ et $T_X \geqslant T_u - \theta$, si θ est le temps de retard de la fonction extrêmale.

2 . Mémorisation

L'analyse d'un système d'équations temporelles de repères conduit à des doubles contraintes, dans le cas des boucles de mémorisation. Ces boucles doivent donc être décelées avant l'évaluation des contraintes.

Il y a une boucle de mémorisation, lorsqu'une expression d'un des deux types suivants est rencontrée :

1) $s \leftarrow \underset{\text{maximale}}{(\text{expression E})} + s. \underset{\text{minimale}}{(\text{expresssion m})}$

2) $S \leftarrow \underset{\text{minimale}}{(\text{expression m})}. (\underset{\text{maximale}}{(\text{ expression E})} + S)$

L'analyse du système d'équations consiste à partir des sorties d'état (effets) pour remonter jusqu'aux signaux d'entrée (causes).Comme chaque équation élémentaire correspond à une fonction extrêmale (f) ($s \leftarrow f(e_1,\ldots,e_n)$),

les variables d'entrée e_i doivent être remplacées par leur expression propre.

Lorsque ces variables sont des variables d'entrée du système,
1) leur état doit correspondre à celui donné par la définition, si ces entrées sont caractéristiques ;
2) lorsqu'il n'en est pas ainsi, ces variables doivent pouvoir être éliminées, comme nous l'avons déjà expliqué.

Lorsque les variables sont des variables intermédiaires, il faut, avant d'effectuer la substitution de leur symbole par l'expression associée, vérifier si cette variable n'a pas déjà été l'objet d'une substitution. Lorsqu'il en est ainsi, un état de mémorisation est détecté. Cette détection doit commencer à partir de la substitution la plus récente pour remonter jusqu'à la plus ancienne. Le cheminement correspond à la remontée de l'arborescence d'analyse d'une expression de sortie.

Lorsque l'imbrication des boucles de mémorisation est décelée, il faut alors déterminer les causes d'établissement et de maintien.

Les contraintes d'établissement doivent d'abord être vérifiées par rapport aux entrées du système : le but est en premier de rechercher des causes directes Relativement à une imbrication de boucles de mémorisation, celà revient à "descendre l'arbre" par les composantes d'établissement.

Lorsque cette descente est impossible - par rapport à la définition de l'état - on en déduit que l'établissement ne peut être seulement dû qu'à une composante de maintien, c'est à dire à un autre état. Autrement dit une succession obligatoire d'états a été décelée. La validité de la cause d'établissement n'est cependant vérifiée que si cette succession est autorisée par la définition des séquences d'états.

Celà met en évidence l'importance d'un énoncé complet, car seul cet énoncé permet de réduire le nombre de solutions possibles.

3 . Succession des états

Nous avons défini, lors de l'étude de la cellule de mémoire, l'algorithme qui permet de vérifier la succession des états. Il consiste à calculer les coordonnées des isochrones par rapport à une sortie d'état et à vérifier :

- 1) que les composantes des sorties d'état sont adjacentes,
- 2) que les composantes des autres signaux se succèdent.

4 . Exemple d'analyse d'un circuit séquentiel

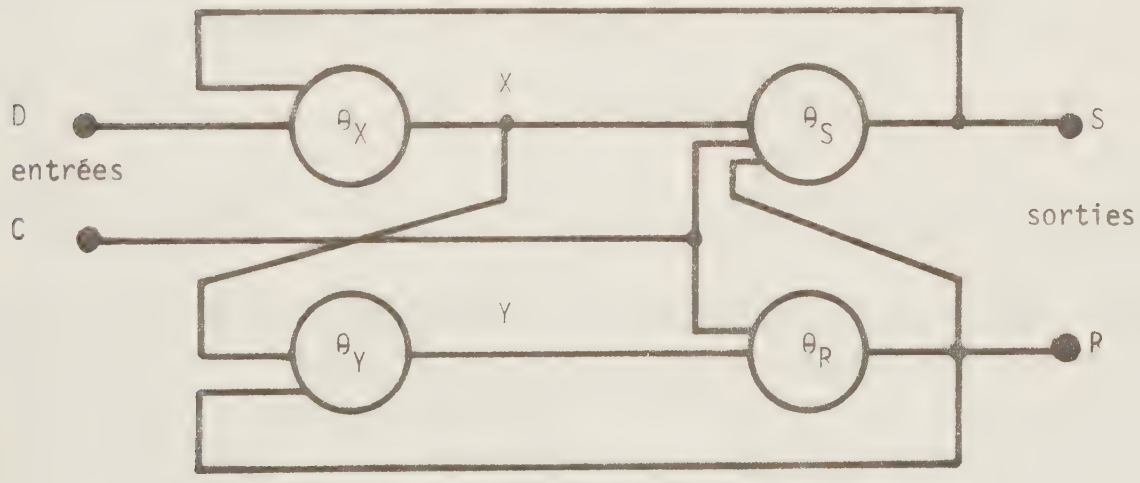
L'exemple donné permet de comprendre la méthode d'analyse descendante des arborescences et la détection des boucles de mémorisation. Le circuit séquentiel est défini à l'aide d'un schéma où les ronds représentent des portes symétriques.

Le problème posé est de savoir si les sorties d'état R et S peuvent avoir la valeur minimale r et maximale S, à la seule condition que la commande C soit dans l'état neutre c (ie : minimal).

L'analyse permet de détecter deux boucles de mémorisation liées entre elles et deux conditions initiales : les états des composantes r et x doivent être prédéfinis. L'étude montre que ces états initiaux ne peuvent être dûs qu'aux composantes extrêmales C et D.

La réponse au problème posé est positive, mais il reste à étudier l'initialisation. Celà revient à vérifier que l'état (C, D, x, r) est un état du système. Les contraintes temporelles sur l'entrée D sont déduites de cette vérification.

L'étude complète de ce système séquentiel est effectuée par la suite, à la fin du chapitre : ce système a une fonction bien précise que nous expliquerons.



Système d'échantillonnage à mémoire

ETAT A = (r, S, c) /ref: (t_c, T_c)

Analyse descendante à partir de la sortie d'état R

r → (θ_R) | C → *n'appartient pas à l'état A*
| +
| Y → (θ_Y) | r → *boucle $r \leftarrow C + r.x$ ⇒ mémorisation établissement: $C \notin$ état A ⇒ r et C appartiennent à l'état précédent*
| .
| x → (θ_X) | D → *n'appartient pas à l'état A*
| +
| S → *sortie d'état*

S → (θ_S) | c → *entrée ⇒ contraintes temporelles*
1) r/c : $\theta_R+\theta_Y+\theta_X+\theta_S$
2) S/c : θ_S
| .
| x → *boucle $x \leftarrow D + x.(c.r)$ ⇒ mémorisation établissement: $D \notin$ état A ⇒ x et D appartiennent à l'état précédent*
| .
| r → *sortie d'état déjà évaluée*

Conclusions de l'analyse:

L' ETAT A EST UN ETAT DU SYSTEME SI IL EST PRECEDE DE L'ETAT (r, D, C)

ETAT B = (r, D, C) /ref: (t_C, T_C)

Analyse descendante à partir de la sortie d'état R

r → (C, +, Y) (θ_R)
- C → *entrée ⇒ contrainte temporelle r/C : θ_R*
- Y → (r, ., x) (θ_Y)
  - r → *boucle r ← C + r.x ⇒ mémorisation établissement: durée de C ⟩ $\theta_R+\theta_Y$*
  - x → (D, +, S) (θ_X)
    - D → *entrée ⇒ contrainte temporelle r/D : $\theta_R+\theta_Y+\theta_X$*
    - S → *sortie d'état*

S → c.x.r → *cas exclu car c n'appartient pas à l'état B ⇒ état s ← C + R + X*

s → (C, +, R, +, X) (θ_S)
- C → *entrée ⇒ contrainte temporelle s/C : θ_S*
- R → *n'appartient pas à l'état B*
- \+
- X → *n'appartient pas à l'état B puisque x fait partie de l'état (cf. sortie R)*

Conclusion de l'analyse: L'ETAT B EST UN ETAT DU SYSTEME

SUCCESSION DES ETATS B = (r, D, C) ET A = (r, S, c)

1) les composantes C et c sont obligatoirement adjacentes

2) Sortie d'état s: état B: $s/C=\theta_S$, état A: $S/c=\theta_S$ ⇒ composantes adjacentes

3) Sortie d'état r: deux solutions car r ← C + Y

3.1) état B: $r/C=\theta_R$,état A: $r/c=\theta_R+\theta_Y+\theta_X+\theta_S$ ⇒ rejet (composantes non adjacentes)

3.2) état B: $r/D=\theta_R+\theta_Y+\theta_X$, état A: $r/c=\theta_R+\theta_Y+\theta_X+\theta_S$ ⇒ contrainte nécessaire sur D: *$D/C=\theta_S$ ⇒ composantes adjacentes*

Conclusion: LES ETATS B ET A PEUVENT SE SUIVRE

QUESTION : *L'état A = (r, S, c) , repéré par rapport aux paramètres (t_c , T_c) est-il un état du système séquentiel?*

REPONSE : *oui, si cet état est précédé de l'état B = (r, C, D, s) tel que ces deux états se succèdent.*
Celà implique que:
1) la composante C a une durée supérieure à $\theta_R + \theta_Y$;
2) la composante D est active sur l'intervalle $[-\theta_X , +\theta_S]$, par rapport au point de transition (C, c) de la commande C.

Lorsque les états élémentaires sont définis à l'échelle de la cellule binaire de mémoire, l'algorithme complet que nous venons d'exposer n'est même pas nécessaire. L'exposé des propriétés des bascules le prouve : les analyses effectuées y sont très élémentaires ; mais, par regroupement, elles permettent d'expliquer complètement le fonctionnement logique de toutes les bascules.

Lorsque la structure d'un système séquentiel devient trop complexe, la solution la plus raisonnable est de redéfinir un"alphabet" dont les caractères sont les séquences d'états de l'étape précédente de l'analyse. La définition de cet alphabet ne pose aucun problème, lorsque la définition de la structure est effectuée graduellement " en partant du haut" - c'est à dire : à partir d'une définition macrocospique de la structure.

B - ECHANTILLONNAGE

I - SIGNAL DE COMMANDE

En logique binaire, l'alphabet associé à n'importe quel signal physique ne peut comporter que deux caractères. Il en résulte que :
- 1) toute boucle de mémorisation ne peut avoir que deux états propres,
- 2) la génération de ces états propres nécessite deux signaux physiques puisqu'un alphabet ternaire est nécessaire :

  - deux caractères d'établissement,
  - un caractère de maintien ("séparateur"),

- 3) tout système qui possède plus de deux états propres est obligatoirement une combinaison de cellules bistables de mémoire. Autrement dit, l'état du système se décompose alors sur un ensemble de doublets d'états propres, où un doublet est associé à chaque cellule de mémoire.

La conception d'un système séquentiel pose alors deux problèmes complémentaires :
- 1) celui de la correspondance d'alphabets binaires et ternaires,
- 2) celui du repérage commun de l'ensemble des états propres.

C'est ici qu'apparaît la notion de signal de commande :

- D'une part, seul un signal supplémentaire permet de transformer un alphabet binaire en alphabet ternaire, par adjonction d'un caractère neutre (séparateur) ;
- d'autre part, un système de repérage commun implique que les caractéristiques temporelles de tous les signaux soient mesurables par rapport à une même référence.

La notion de signal de commande est incluse à la fois dans le concept d'état - par sa fonction de "séparateur" - et dans celui de temps - par sa fonction de référence temporelle.

Relativement à la mémorisation, un signal de commande a pour fonction de

de définir - statiquement et temporellement - les "échantillons" des signaux d'entrée qui constituent:

- ou bien la cause d'établissement,
- ou bien la cause de maintien d'un état propre.

Le système d'échantillonnage associé à un signal de commande comporte alors:

- une entrée de "données" : D,
- une entrée de commande : C,
- deux sorties d'état : Q et Q'.

L'"alphabet" associé à ce système comporte trois "caractères" (ie: états):

- neutre (e) : (q, q'),
- "SET" (S) : (D, Q, q'),
- "RESET" (R) : (d, q, Q').

Le système d'échantillonnage doit alors "accepter" les deux successions :

- (e, S, e),
- (e, R, e).

II - CARACTERISTIQUES D'UN SYSTEME D'ECHANTILLONNAGE

2.1. Séparateurs

L' "injection" de caractères neutres (séparateurs) est réalisée à l'aide de la fonction de porte logique. De plus, comme nous supposons que tous les alphabets sont binaires, cette injection implique un dédoublement du signal d'entrée D en deux signaux "complémentaires", afin d'obtenir un alphabet ternaire. L'injection de séparateurs résulte donc de la composition de deux types de fonctions: celui de porte et celui de complémentation. Quelquesoit le schéma de composition, la fonction de complémentation doit précéder celle de porte logique, afin d'obtenir un caractère séparateur neutre en sortie.

2.2. Référence de temps

A chaque caractère de sortie (e,R,S) peut être associée une isochrone

dont les paramètres (t_s, d_s) sont fonctions des paramètres des signaux d'entrée D et C.

Contrairement à la cellule de mémoire, pour laquelle chacun des deux paramètres t et d dépend d'un signal différent, les deux paramètres sont maintenant fonctions d'un même signal : le signal de commande C. De même, les contraintes temporelles sur le signal d'entrée D s'expriment par rapport à ce signal de commande C qui sert ainsi de référence de temps.

2.3. Classification

Relativement au signal de commande C, les sytèmes d'échantillonnage peuvent être classés de la façon suivante :

- 1) échantillonnage permanent : la durée de l'isochrone de sortie est fonction de celle des deux isochrones d'entrée D et C ;
- 2) échantillonnage temporaire : la durée de l'isochrone de sortie est un invariant du système ;
- 3) échantillonnage mémoriel : la durée de l'isochrone de sortie ne dépend uniquement que de la durée de la commande C.

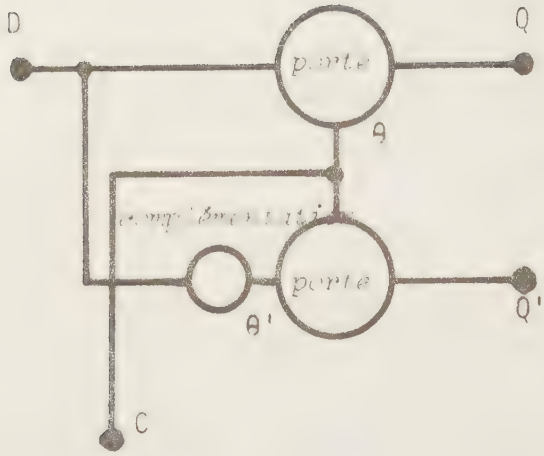
Par définition du système de repérage, la date de l'isochrone de sortie n'est fonction que de la date de l'isochrone C. La date d'entrée D est fixée par rapport à celle de l'isochrone C , la relation obtenue constitue une contrainte temporelle sur l'entrée de donnée, par rapport à la référence C.

III - ECHANTILLONNAGE PERMANENT

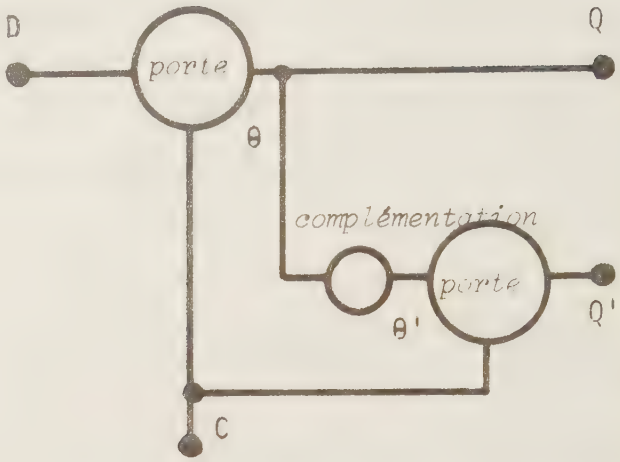
Ce type d'échantillonnage est transparent pour l'entrée D. C'est à dire que pour un même état neutre de la commande C, il y a autant d'états (ie : "caractères") de sortie qu'en comporte l'entrée D sur une période de temps définie par la durée de l'état neutre de la commande C . Ce principe d'échantillonnage est celui des circuits analogiques "track and hold".

La structure d'un système d'échantillonnage permanent est la plus simple qui soit ; elle comporte une fonction de complémentation et deux portes logiques. Deux schémas de composition de ces fonctions sont alors possibles, relativement à la position de la fonction de complémentation :

- complémentation en "amont",
- complémentation en "aval".



complémentation en amont

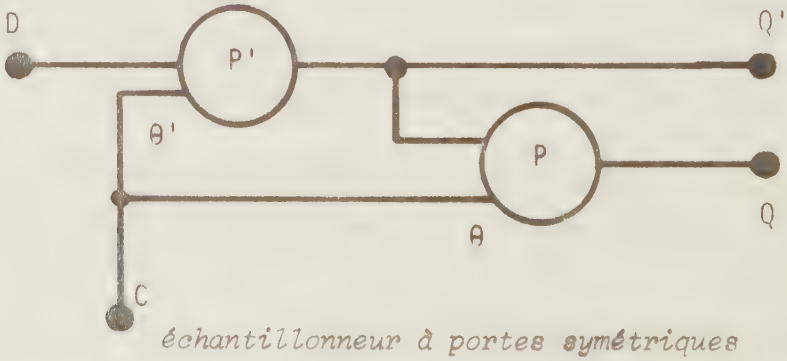


complémentation en aval

D'après la définition que nous en avons donnée dans le chapitre précédent, les portes logiques de ces schémas ne sont pas symétriques : elles sont commandées uniquement par la commande C.

Ainsi, lorsque l'état de la commande C est neutre, il y a une bijection entre les alphabets associés aux signaux D et (Q, Q'). Par contre, lorsque la commande C est active, les signaux Q et Q' sont dans l'état neutre.

Le schéma "aval" peut être remplacé par la composition de deux portes symétriques, où les signaux D (resp. Q') et C ont une action symétrique sur la porte P' (resp. P) :



échantillonneur à portes symétriques

Si les deux schémas "aval" et "amont" ont les mêmes propriétés statiques, ils n'ont pas, par contre, les mêmes propriétés temporelles. Pour mettre cette différence en évidence, nous calculons pour chaque schéma les coordonnées des isochrones e, S, R - par rapport aux paramètres du signal de commande - et étudions les successions (e, S, e) et (e, R, e).

3.1. Complémentation en amont

Les équations des repères sont les suivantes :

1) Sortie d'état Q : état maximal : $Q \leftarrow D.c$
état minimal : $q \leftarrow C + d.c$

2) Sortie d'état Q' : état maximal : $Q' \leftarrow (d).c$
état minimal : $q' \leftarrow C + (D).c$

Les coordonnées des isochrones (e, S, R) , par rapport à la commande C, sont alors :

ioschrone e : (C, q, q')

C		0 ; 0
q	=	θ ; θ
q'		θ'; θ'

$t_C ; T_C$

isochrone S : (D, c, Q, q') avec $Q \leftarrow D.c$ et $q' \leftarrow \overline{D}.c$

c		0 ; 0
D	=	$\leqslant 0$; $\geqslant 0$
Q		θ ; θ
q'		θ'; θ'

$t_C ; T_C$

isochrone R : (d, c, q, Q') avec $q \leftarrow d.c$ et $Q' \leftarrow \overline{d}.c$

c		0 ; 0
d	=	< 0 ; $\geqslant 0$
q		θ ; θ
Q'		θ'; θ'

$t_C ; T_C$

Ces trois isochrones sont repérées par rapport au même signal C. Ce signal de commande est supposé partout défini , ce qui implique que l'union des repères minimal c et maximal C (C + c) est un repère complet. Autrement dit les bornes t_C (resp. t_c) et T_c (resp. T_C) sont confondues.

Il en résulte que, pour les séquences (e, S, e) et (e, R, e), les composantes des sorties d'état sont adjacentes. Les successions (e,S,e) et (e,R,e) sont donc "acceptées" dans le cas de la complémentation en "amont".

3.2. Complémentation en aval

Les équations des repères deviennent alors :

1) Sortie d'état Q : Portes non symétriques : $Q \leftarrow D.c$; $q \leftarrow C + d.c$
Portes symétriques : $Q \leftarrow q'.c$; $q \leftarrow C + Q'$

2) Sortie d'état Q' : Portes non symétriques : $Q' \leftarrow (\overline{q}).c$, $q' \leftarrow C + (\overline{Q}).c$
Portes symétriques : $Q' \leftarrow d.c$, $q' \leftarrow C + D$

Les coordonnées des isochrones (e,S,R) par rapport à la commande C deviennent alors :

Isochrone e
(C, q, q')

C	=	0; 0
q		θ; θ
q'		θ'; θ'

C

Isochrone S :
(D, c, Q, q')
avec $\begin{cases} Q \leftarrow D.c \\ q' \leftarrow \overline{Q}.c \end{cases}$ ou $\begin{cases} q'.c \\ D \end{cases}$

c	=	0; 0
D		⩽ 0; ⩾ 0
Q		θ; θ
q'		θ + θ'; θ'

c

;

c	=	0; 0
D		- θ' ⩽ ; - ⩾ θ'
Q		θ; θ
q'		0; 0

c (1)

ou bien

0; 0	
0; 0	
θ' + θ; θ	
θ'; θ'	

c (2)

(portes non symétriques) (portes symétriques)

Remarquons que dans le cas des portes symétriques, les coordonnées de la sortie d'état q' ne dépendent pas de la commande C, mais uniquement de celles de l'entrée de donnée D. Les contraintes temporelles sur D sont alors calculées de telle façon que les composantes de sortie q' soient adjacentes pour les isochrones S et e, puiqu'elles doivent se succéder (coordonnées (2)).

Isochrone R :
(d, c, q, Q')
avec $\begin{cases} q \leftarrow d.c \\ Q' \leftarrow \overline{q}.c \end{cases}$

c	=	0; 0
d		⩽ 0; ⩾ 0
q		θ; θ
Q'		θ + θ'; θ'

c

ou bien

c	=	0; 0
d		⩽ 0; ⩾ 0
q		θ' + θ; θ' + θ
Q'		θ'; θ'

c

(portes non symétriques) (portes symétriques)

Quel que soit le type des portes logiques, les trois isochrones (e, S, R) appartiennent donc à l'"alphabet"du système d'échantillonnage.

De même, les successions (S, e) et (R, e) sont "acceptées", puisque les composantes de sortie sont adjacentes (ie : leurs coordonnées par rapport au point de repère $T_c = t_c$ sont identiques).

Par contre, les successions (e, S) et (e, R) sont "refusées" car les composantes d'une sortie d'état ne sont pas jointives par rapport au point de repère $t_c = T_c$. Si les états (ie : isochrones) e et S (ou e et R) se suivent, il y a obligatoirement un état intermédiaire X entre eux : (e, X, S) ou (e, X, R). Cet état intermédiaire X est parfois appelé aléa .
Il importe de savoir si cet état intermédiaire modifie ou non la succession des composantes des sorties d'état. Lorsque cette succession n'est pas modifiée, l'état intermédiaire ne diffère d'un des deux états "normaux" e ou S (ou R) que par ses causes ; l'effet est alors le même - d'un point de vue statique.

Quel que soit le type des portes logiques, il y a toujours une des deux sorties d'état pour laquelle les composantes sont jointives. Pour étudier l'effet de l'état intermédiaire, il suffit d'étudier la succession des composantes pour l'autre sortie d'état. Nous avons ainsi les successions suivantes :

suite (e, X, S) : 1) portes non symétriques : sortie Q', suite (q', X, q')
2) portes symétriques : sortie Q, suite (q, X, Q)
suite (e, X, R) : 1) portes non symétriques : sortie Q', suite (q', X, Q')
2) portes symétriques : sortie Q, suite (q, X, q)

Il y a deux types de suites, suivant que l'état final est identique ou différent de l'état initial :

-1) suite de type (m, X, m),
-2) suite de type (m, X, M).

Comme les alphabets sont binaires, l'effet de l'état intermédiaire est indiscernable - statiquement - pour le second type de suite :

(m, M) = (m, m, M) ou (m, M, M).

L'effet n'est que temporel : seule la date de transition ($T_m = t_M$) est modifiée.

Par contre, il n'en est pas de même pour le premier type de suite, dans le cas (m, M, m). Il faut alors déterminer l'état X de la sortie, à l'aide des équations temporelles. Nous prenons pour exemple la suite (q', X, q') dans le cas où les portes ne sont pas symétriques :

$$Q' \leftarrow \overline{q}.\ c \qquad \text{d'où} \qquad t_{Q'} = \theta' + \max\ (t_{\overline{q}}\ ,\ t_c)$$
$$q' \leftarrow \overline{Q}.\ c \qquad\qquad\quad\ \ tq' = \theta' + \max\ (t_{\overline{Q}}\ .\ t_c)$$

La composante q due à l'état e existe sur l'intervalle de temps $[T_C,\ T_C + \theta]$, si la durée de la commande C est supérieure à θ.
Comme $T_C = t_c$, il en résulte que la composante Q' existe sur l'intervalle $[\,t_c + \theta',\ t_c + \theta' + \theta\,]$. La séquence de sortie est donc (q', Q', q') : l'état intermédiaire perturbe ainsi la sortie. La séquence d'états devient donc :

(e, R, S) au lieu de (e,S)

Bien que la sortie soit modifiée, l'échantillonnage a le même effet sur la cellule de mémoire, puisque la dernière cause d'établissement est la même.

3.3. Contraintes temporelles

Les contraintes temporelles déterminent l'intervalle de temps pendant lequel l'état du signal de données D doit être stable et celà, par rapport aux paramètres temporels (t_c, Tc) de la commande C. Ces contraintes ont été évaluées lors du calcul des coordonnées des isochrones. Dans tous les cas, il faut que la composante définie (ie : stable) du repère D + d inclut la composante neutre de la commande C (ie : $t_D \leqslant t_c$, $T_D \geqslant T_c$).

Lorsque la complémentation a lieu en "amont" des portes logiques, cette contrainte doit être modifiée si l'on suppose que la complémentation introduit un retard propre θ'_c , distinct de celui qui est introduit par la porte P' (θ'_P). Les contraintes temporelles sont alors les suivantes:

$$t_D \leqslant t_c - \theta'_c \ ; \quad T_D \geqslant T_c$$

La durée (d_S) des caractères de sortie est uniquement fonction de celle de la commande C si les contraintes temporelles précédentes sont vérifiées, c'est à dire si $d_D \geqslant d_c$. Nous avons alors les relations suivantes:

montage "amont" : si $d_D > d_c + \theta'_c$ alors : $d_S = d_c$

montage "aval" : si $d_D > d_c$ alors : $d_S = d_c - \theta'$ portes non symétriques

ou $d_S = d_c - \theta$ portes symétriques

Dans tous les cas, la durée d_S doit être supérieure à la durée minimale d'une composante d'établissement d'un état propre de la cellule de mémoire, c'est à dire $d_S \geqslant \theta_m + \theta_E$.

IV - ECHANTILLONNAGE TEMPORAIRE

La durée δ des échantillons est alors un <u>invariant</u> du système d'échantillonnage. Il en résulte que leur date dépend

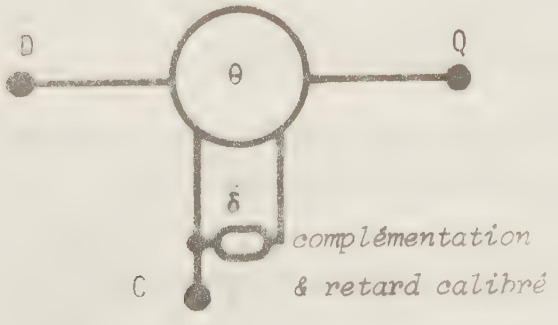
- ou bien du "front avant" $(T_C = t_c)$
- ou bien du "front arrière" $(T_c = t_C)$

du signal de commande C. Autrement dit, l'échantillonnage peut être localisé par rapport à un <u>seul</u> point de transition de la commande C.

Les coordonnées des isochrones associées sont alors calculées par rapport à cette date de transition et à l'invariant δ.

4.1. Echantillonnage sur " front avant".

Le principe est d'ouvrir les portes logiques pendant un laps de temps invariant, à partir du "front avant" de la commande C. Celà revient à <u>refermer</u> les portes logiques un temps invariant après le début de la commande d'ouverture. Il suffit pour celà de commander les portes à l'aide de <u>deux</u> signaux de commande dont l'un est le "complémentaire" <u>retardé</u> de l'autre :



porte à double commande

Le fonctionnement d'une porte logique à double commande vérifie les équations de repères suivantes :

$$Q \leftarrow D.(c.c')$$
$$q \leftarrow d.(c.c') + C + C'$$
$$c' \leftarrow C \text{ et } C' \leftarrow c$$

Les repères C + C' et c.c' sont mutuellement exclusifs et complémentair
Les coordonnées temporelles d'une composante définie du repère c.c' peuvent êtr
définies par rapport à la date de début (t_c) de la composante c :

$$t_{c.c'} = t_c \quad ; \quad T_{cc'} = t_c + \delta$$

Les coordonnées des isochrones associées aux états e, S, R sont alors ca
culées par rapport :
- 1) à la date de début (ie : "front avant" t_c) de l'état neutre de la command
$t_c = T_C$,
- 2) à l'invariant δ du système d'échantillonnage.

a) Etat neutre e (q) : $t < t_c + \theta \quad ; \quad t \geqslant t_c + \theta + \delta$
b) Etat "Set" (D, Q) : - date : $t_Q = t_c + \theta$
- durée : δ
- contraintes : 1) $d_c > \delta$; 2) $t_D \leqslant t_c$, $T_D \geqslant t_c +$
c) Etat "Reset" R (d,q): - date : $tq = t_c + \theta$
- durée δ
- contraintes : $t_d \leqslant t_c$, $T_d \geqslant t_c + \delta$

Il y a deux contraintes temporelles :
- 1) la durée de l'état neutre de la commande doit être supérieure à δ
- 2) l'entrée de donnée D doit être stable sur l'intervalle $[t_c, t_c + \delta[$.

Les états (e, R, e)ou bien (e, S, e) se succèdent. Ce système d'échantillonnage ne présente donc aucun "aléas".

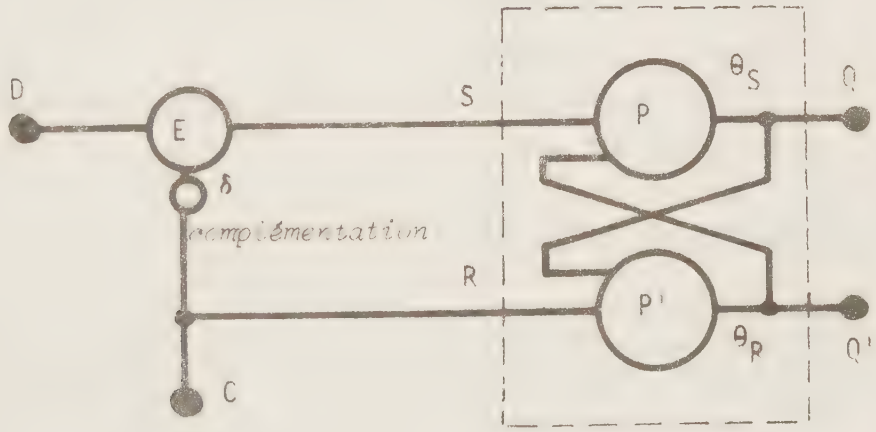
Le système complet d'échantillonnage comprend alors deux portes à double commande et une fonction de complémentation.

4.2. Echantillonnage sur "front arrière"

Le principe de fonctionnement est ici entièrement différent ; celà est dû au fait que l'échantillonnage a lieu " à la fin". Pour que l'effet suive sa cause, la commande C ne doit plus alors être considérée comme la référence de temps associée à l'établissement d'un état propre de la cellule de mémoire, mais au contraire au maintien d'un tel état.

Rappelons que pour une boucle de mémorisation, les composantes définies du repère de maintien ont un état neutre ; l'état actif de ces composantes correspond alors à un état forcé ; la cause de maintien devient alors une cause de "verrouillage" de la cellule.

La fonction de maintien de la commande C Implique que celle-ci agisse directement sur la cellule de mémoire ; il en résulte la structure dissymétrique suivante :



échantillonnage sur "front arrière"

Fonctionnement

1 - Lorsque la commande C est à l'état neutre, la porte d'entrée E est fermée et l'état d'entrée de la cellule de mémoire est neutre (e : (s,r)).

2 - Lorsque la commande C est active, la sortie Q' est forcée à l'état neutre, la porte E est ouverte et l'état de la sortie Q dépend alors de l'entrée D.

Lorsque l'entrée D est active (ie : S puisque la porte non symétrique E

est ouverte), la cellule de mémoire est dans un état forcé (q, q').
Par contre, lorsque l'entrée D est neutre, la cause d'établissement (s, R) de l'état "Reset" est vérifiée.

3 - Lorsque la commande C passe de l'état actif à l'état neutre, la cause de maintien (r) apparaît immédiatement sur la porte P'.

Etat "Set" : nous avons vu que la composante d'établissement doit rester stable, après l'apparition de la composante de maintien, pendant un temps supérieur au retard θ_R, afin de "verrouiller" la boucle. Autrement dit, l'établissement de l'état "Set" nécessite :
- a) que l'entrée S soit active sur l'intervalle de temps $[\ T_C - \theta_S,\ T_C + \theta_R\ [$;
- b) que la porte d'entrée E doit se fermer avec un temps de retard supérieur à θ_R, par rapport à l'instant T_C.

Etat "Reset" : L'établissement de l'état "Reset" (s,R) suppose par contre
- a) que la durée de l'état actif de la commande C soit supérieure à la période $\theta_R + \theta_S$;
- b) que l'entrée S soit à l'état neutre (s) au moins à partir de la date $(T_C - \theta_S)$.

Caractéristiques temporelles

Les différents repères ont pour expression :

$S \leftarrow D.C$; $s \leftarrow d.\ C + c$
$R \leftarrow C$; $r \leftarrow c$

Les coordonnées des isochrones associées aux états e, S, R sont alors

Etat neutre e (r, s) : frontière supérieure $\begin{matrix} r \\ s \end{matrix}\Big|\begin{matrix} 0 \\ \delta \end{matrix}$; frontière inférieure $\begin{matrix} r \\ s \end{matrix}\Big|\begin{matrix} 0 \\ \delta \end{matrix}$

t_C T_C

Etat "Set" S (D,S,r) : $\begin{matrix} D \\ S \\ r \end{matrix}\Big|\begin{matrix} \leqslant 0;\ \geqslant 0 \\ \delta\ ;\ \delta \\ -\ ;\ 0 \end{matrix} = \begin{matrix} -\ d_C,\ 0 \\ \delta - d_C,\ \delta \\ 0,\ d_c \end{matrix}\Big|$

t_C; T_C ($d_D \geqslant d_C$) T_C

Etat "Reset" R (d, s, R) :

$$\begin{array}{l|c|l} d & & 0;\ 0 \\ s & = & \delta;\ \delta \\ R & & 0;\ 0 \end{array} \quad = \quad \begin{array}{l} -d_C,\ 0 \\ \delta - d_C,\ \delta \\ -d_C,\ 0 \end{array}$$

$$t_C;\ T_C\ (d_d \geqslant d_C)\quad T_C$$

Les contraintes temporelles, tirées de l'étude de la cellule RS de mémoire, sont les suivantes :

1) Retard : $\delta \geqslant \theta_R$

2) Commande C : $d_C \geqslant \theta_R + \theta_S$

3) Donnée D : $t \leqslant T_C - (\theta_R + \theta_S)$; $T \geqslant T_C$

Si l'on suppose que l'entrée de donnée D est stable lorsque la commande C est active, alors les états (e, R, e) se succèdent. Par contre, l'état S ne peut succéder à l'état e, puisque l'état r n'apparait que lorsque la commande C redevient neutre. D'après la valeur des coordonnées, nous voyons que l'état S est précédé de la succession (R^*, F) où les états R^* et F ont pour coordonnées :

$R^* = (s, R)$:

s	0 ; 0
R	0 ; 0

t_C ; δ

; F = (D, S, R) :

D	0 ; 0
S	δ ; δ
R	0 ; 0

t_C ; T_C

La succession d'états associée à l'établissement de l'état "Set" est donc :

(e, R^*, F, S, e)

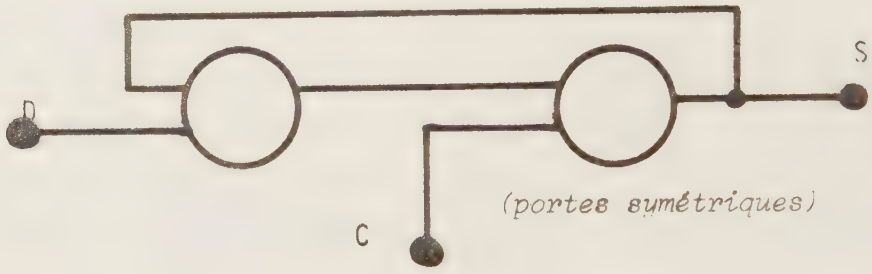
L'état R^* correspond à l'établissement de l'état "Reset" ; il peut être qualifié d'"aléas" dans la mesure où il n'est pas discernable dans l'autre succession (e, R, e)

V - ECHANTILLONNAGE MEMORIEL

Le principe de ce type d'échantillonnage est de découpler la cause (ie : le signal échantillonné D) de son effet (ie : l'établissement d'un état propre de la cellule de mémoire).

Pour ce faire, le signal est échantillonné sur le "front avant" de la commande et est mémorisé pendant la durée de celle - ci.

Le fondement du principe est l'utilisation d'une cellule de mémoire F en "régime forcé".
- La sortie d'état S de cette cellule F correspond à une entrée de la cellule "effective" de mémoire RS ;
- Le signal échantillonné D est la cause d'établissement de l'état propre "actif" S de la cellule F ; alors que la commande C en est la cause de maintien, ou bien au contraire est la cause d'établissement de l'état neutre de la sortie S (régime forcé) :



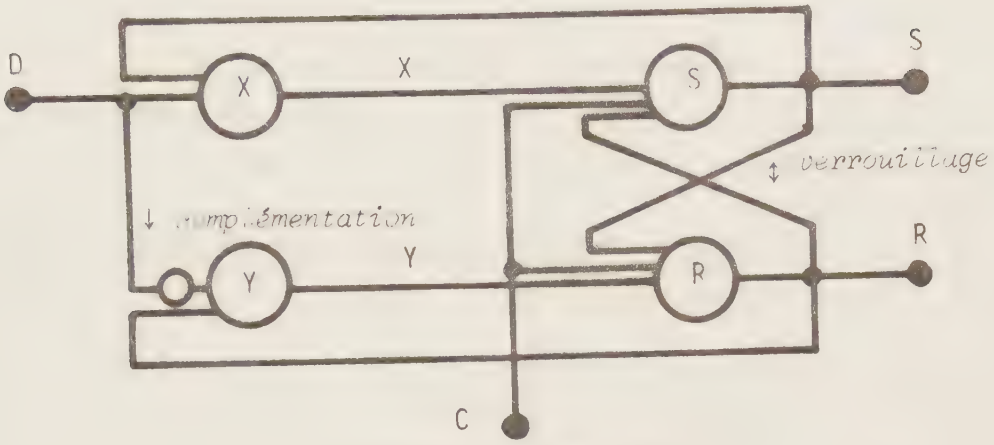
cellule F d'échantillonnage

- 1 Lorsque la commande C est active, la sortie S est à l'état neutre (la commande C "verrouille" la cellule F) ;
- 2 Lorsque la commande C est à l'état neutre, l'établissement de l'état actif de la sortie S est du à la première composante active du signal d'entrée D. Une fois établi, cet état propre de la cellule F est indépendant de l'état de l'entrée D : il y a alors découplage entre l'entrée D et la sortie S.

Le système complet d'échantillonnage comporte deux cellules de ce type et une fonction de complémentation en "amont".

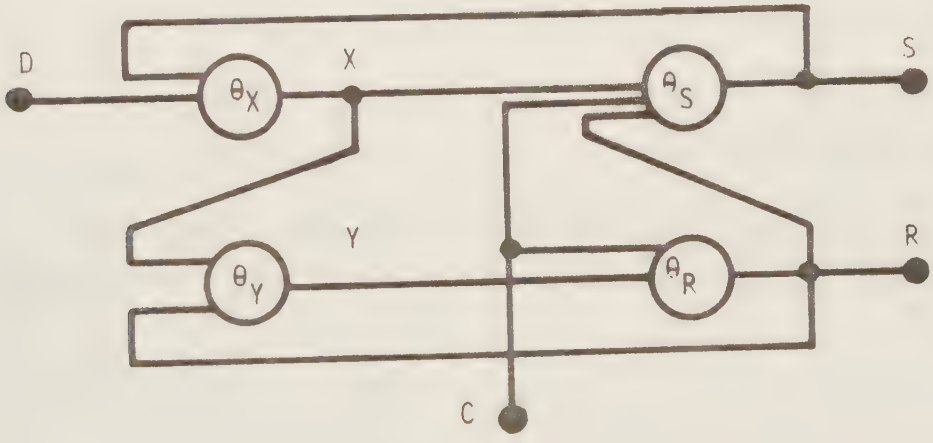
Lorsque l'état actif d'une de ces deux cellules est établi, cet état ne dépend plus du signal d'entrée D. Par contre, l'établissement du même état pour l'autre cellule peut avoir lieu si le signal d'entrée change d'état. Pour empêcher cet établissement, la première cellule doit alors verrouiller la seconde.

Le schéma du système complet est alors le suivant :



échantillonnage à mémoire

Comme les portes symétriques ont un effet "inverseur", la fonction de complémentation peut être "économisée", grâce au schéma suivant. Le verrouillage de la cellule R n'est plus nécessaire, puisque la sortie S verrouille la porte d'entrée X du système.



structure réduite

5.1. Caractéristiques temporelles

a) CELLULE D'ECHANTILLONNAGE

Les différents repères ont pour expression :

- 1) $S \leftarrow x.c$, $x \leftarrow D + S$; d'où $S \leftarrow (D.c) + S.(c)$ (mémorisation)
l'établissement de l'état actif S implique que l'entrée D soit active et que la commande c soit à l'état neutre.

- 2) $s \leftarrow C + X$, $X \leftarrow d.s$; d'où $s \leftarrow (C) + s.(d)$ (mémorisation) :
l'état s est maintenu en l'absence de la composante C si l'entrée est à l'état neutre. Cet état ne peut être établi que par la commande C.

De ces expressions, nous tirons les coordonnées des isochrones associées aux états (s, C), (d, s, c) et (D, S, c) :

Etat forcé (s, C) :

	t_C	T_C
s	θ_S	θ_S

Etat neutre (d, s, c) : (précédé de l'état forcé)

	t_c	T_c
d	$-\theta_X$	$-\theta_X$
X	0	0
s	θ_S	θ_S

Etat actif (D, S, c) : état propre si $d(D.c) \geqslant \theta_S$

	t_c	T_c
D	$(-\theta_X, \theta_S)$	
x	0	0
S	θ_S	θ_S

b) STRUCTURE REDUITE

Les différents repères ont maintenant pour expression :

$$S \leftarrow (D.(c.r)) + S.(c.r) \quad ; \quad s \leftarrow (C + R) + s.(d)$$
$$R \leftarrow (d.s.c) + R.(c) \quad ; \quad r \leftarrow (C) + r.(D + S)$$

L'analyse de ces équations a été donnée en exemple précédemment.
Les coordonnées des isochrones associées aux "échantillons" ont les valeurs suivantes :

Etat neutre e (s, r) :

	t_C	T_C
s	θ_S	θ_S
r	θ_R	θ_R

Etat "Set" S (D, S, r) :

	t_c	T_c
D	$(-\theta_X, \theta_S)$	
S	θ_S	θ_S
r	θ_R	θ_R

Etat "Reset" R (d, s, R) . Afin de déterminer les contraintes temporelles sur l'entrée D, les coordonnées de l'état (X, R) sont d'abord calculées :

$$\begin{array}{l|l} X & (-\theta_Y, \theta_R); - \\ R & \theta_R \ ; \ \theta_R \end{array}$$

$t_c ; T_c$

Puis les coordonnées de l'état (d,X, s)

sont ensuite calculées par rapport au repère (c.r) , de façon que la composante X soit définie sur l'intervalle $[\, t_c - \theta_Y \ ; \ t_c + \theta_R \,[$:

$$\begin{array}{l|l} d & (-\theta_X - \theta_Y, \theta_R - \theta_X) \ ; - \\ X & (-\theta_Y, \theta_R) \ ; - \\ s & \theta_S \ ; \ \theta_S \end{array}$$

$t_{c.r} ; T_{c.r}$

Les coordonnées de l'isochrone R minimale sont donc :

$$\begin{array}{l|l} d & (-\theta_X - \theta_Y, \ \theta_R - \theta_X) \ ; - \\ s & \theta_S \ ; \ \theta_S \\ R & \theta_R \ ; \ \theta_R \end{array}$$

$t_c ; T_c$

Il y a ainsi deux contraintes temporelles pour les états R et S :

- 1) la durée de l'état neutre de la commande C doit être supérieure à $\max(\theta_S, \theta_R)$ ("bouclage" des cellules de mémoire, à travers les portes S et R , pour la mémorisation des états actifs),

- 2) l'entrée de donnée D doit être stable sur l'intervalle de temps $[\, t_c - \theta_X - \theta_Y \ ; \ t_c + \max(\theta_S, \theta_R - \theta_X) \,[$

(cette contrainte garantit - 1) la position des composantesd'établissement par rapport à celle de maintien c et - 2) la stabilité de ces composantes).

D'autre part, les états (e, S, e) ou bien (e, R, e) se succèdent, il n'y a donc pas d'"aléas".

C - TRANSPARENCE

Nous venons d'étudier différentes structures d'une cellule de mémoire binaire. Dans tous les cas, la cellule comporte :

- une entrée de donnée ;
- une entrée de commande,
- une ou deux sorties d'état,
- un système d'échantillonnage,
- une boucle "RS" de mémorisation.

Nous avons défini la mémorisation comme une opération de "décalage" d'un repère d'entrée sur un repère de sortie. Les composantes définies de ces deux repères délimitent les intervalles de temps pendant lesquels les signaux d'entrée ou de sortie sont stables.

Le système d'échantillonnage permet de définir la fonction de mémorisation par rapport au signal de commande. Ainsi, le repère de sortie est déduit du repère associé au signal de commande, à condition que le repère d'entrée vérifie certaines relations temporelles par rapport à ce même repère de commande. Ces relations sont des relations d'inclusion : le repère d'entrée doit inclure un repère minimal déduit du repère de commande.

Le repère d'entrée caractérise les composantes d'établissement des deux états propres de la cellule de mémoire, alors que le "complément" temporel du repère de commande caractérise la cause de maintien de ces états propres.

La commande simultanée de telles cellules de mémoire pose alors un problème temporel,lorsque l'établissement de l'état propre d'une cellule dépend de celui d'une autre cellule. Il faut en effet que la cause d'établissement soit stable, alors que la commande peut déclencher un changement d'état - donc un état instable - de toutes les cellules !

La seule solution, pour que ce problème ne se pose pas - quelque soit la combinaison des cellules - est que la commande d'un changement d'état de l'ensemble des cellules soit telle que la fin des causes minimales d'établis-

sement doit précéder le début des effets, c'est à dire des changements d'état des cellules.

Autrement dit, tout repère d'entrée "conditionnelle" (ie : l'initialisation est exclue) doit pouvoir être inclus dans n'importe lequel des repères de sortie , quelles que soient les valeurs des paramètres temporels des cellules.

Nous dirons que les cellules de mémoire qui ont cette propriété sont opaques.

Aucune des cellules étudiées n'a cette propriété, comme le montre l'examen des coordonnées des isochrones, pour les sorties d'état.

1 - Pour l'échantillonnage permanent ou temporaire, l'entrée de donnée D doit être stable jusqu'à ce que la cellule RS de mémoire soit positionnée. Ce type de cellule est dit transparent.

2 - Il n'en est pas tout à fait de même pour l'échantillonnage mémoriel. Ainsi, pour chacune des causes d'établissement (état "Set" ou état "Reset") la stabilité de l'entrée n'est plus imposée à partir du moment (date $t_c + \theta_S$ ou $t_c + \theta_R$) où les composantes R ou S apparaissent.

Cependant, les contraintes temporelles ne sont pas les mêmes pour les deux états : - ainsi pour l'état "Set" : $T_D \geqslant t_c + \theta_S$, alors que $t_S = t_c + \theta_S$;

- et pour l'état "Reset" : $T_d \geqslant t_c + \theta_R - \theta_X$, alors que $t_R = t_c + \theta_R$.

Une même contrainte temporelle sur l'entrée D ne permet de vérifier la mutuelle exclusion de la donnée et de l'échantillon que si les temps de retard sont liés par la relation $\theta_S \leqslant \theta_R$. Malheureusement, la date limite de stabilité ($t_c + \theta_S$) de l'entrée de donnée D dépend de la cellule, à cause du temps de retard θ_S. Si les cellules n'ont pas les mêmes paramètres temporels, la propriété requise n'est alors pas vérifiée, bien qu'elle le soit pour chaque cellule prise séparément.

I - LES CELLULES A "DOUBLE - PONT"

Puisqu'une simple cellule de mémoire ne peut être "opaque", la solution consiste alors à mettre deux cellules de ce type en "cascade" (ie : double - pont) et à les commander de telle façon que le repère d'entrée de l'une soit

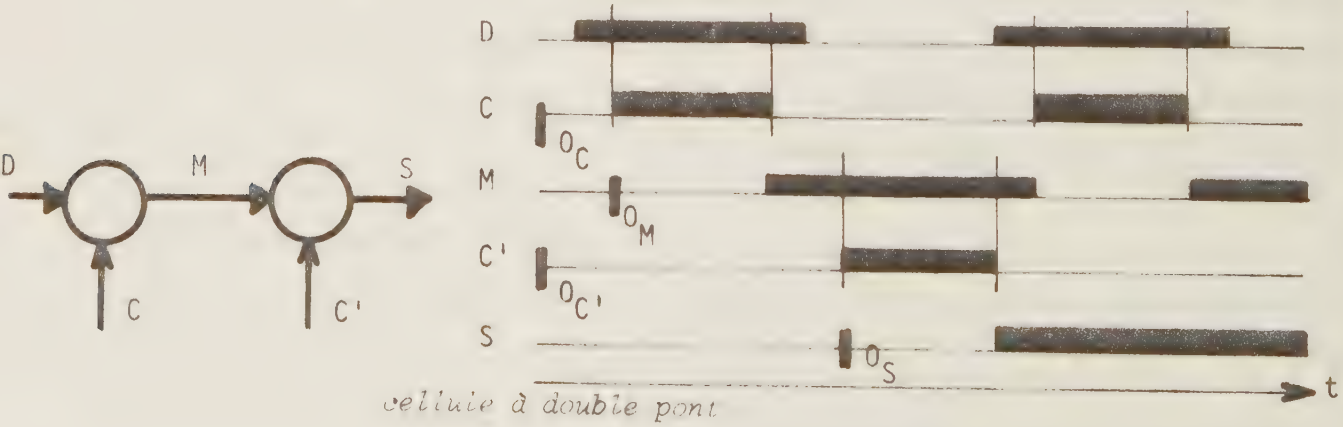
inclus dans le repère de sortie de l'autre. Il faut de plus que cette double relation d'inclusion soit vérifiée quelle que soit la valeur comparée des paramètres temporels des cellules, afin que l'on puisse combiner des cellules doubles de ce type de n'importe quelle façon.

La seule manière de garantir cette relation d'inclusion, pour un ensemble quelconque de cellules doubles, est de l'exprimer par rapport aux signaux de commande, sans faire référence aux paramètres temporels de chaque cellule. Autrement dit, nous avons là un problème de repérage purement qualitatif. Celà n'exclut pas que les signaux de commande doivent vérifier certaines contraintes temporelles, de façon que les éléments physiques assument leur fonction logique.

Pour que la propriété d'opacité soit vérifiée qualitativement, chaque cellule de base d'une cellule double doit assumer la fonction logique de mémorisation, telle que nous l'avons définie par rapport au signal de commande:

1) le repère de commande doit être inclus dans le repère d'entrée, à la limite ces deux repères peuvent être identiques;
2) le repère de sortie doit être le"mémorisé" (ie: complémentaire temporel) du signal de commande;
3) il n'y a qu'une contrainte temporelle: la durée des composantes du repère de commande doit être supérieure au plus grand temps d'établissement d'un état propre.

La cellule double de mémoire est alors telle que le repère de commande de chaque cellule est inclus dans le repère de sortie de l'autre (nous excluons le problème de l'initialisation).



cellule à double pont

II - LES BASCULES "MAITRE - ESCLAVE".

A la limite, l'union des repères C + C' peut être - logiquement - un repère complet. Les deux commandes C et C' peuvent alors être véhiculées par un même signal physique dont l'alphabet associé est binaire.

En fait, cet alphabet doit être ternaire, à cause de la dispersion des caractéristiques physiques des cellules de base. En effet, un seul et même point de transition distingue la fin d'une commande (C) et le début de l'autre (C'). Il se peut alors que le début d'une commande soit détecté avant la fin de l'autre, puisque ces deux commandes ne sont pas destinées à la même cellule. Tout dépend, dans ce cas, des caractéristiques des fonctions de seuils de chaque cellule. La fin d'un état stable et le début de l'état stable complémentaire doivent donc être discernés séparément par chaque cellule. Celà nécessite la définition d'un "double seuil" pour la grandeur caractéristique du signal qui véhicule les deux commandes.

Relativement au signal de commande, les systèmes d'échantillonnage des deux cellules de base fonctionnent de façon complémentaire, leurs fonctions de seuil ne détectent - pour chaque cellule - qu'un seul des deux seuils.

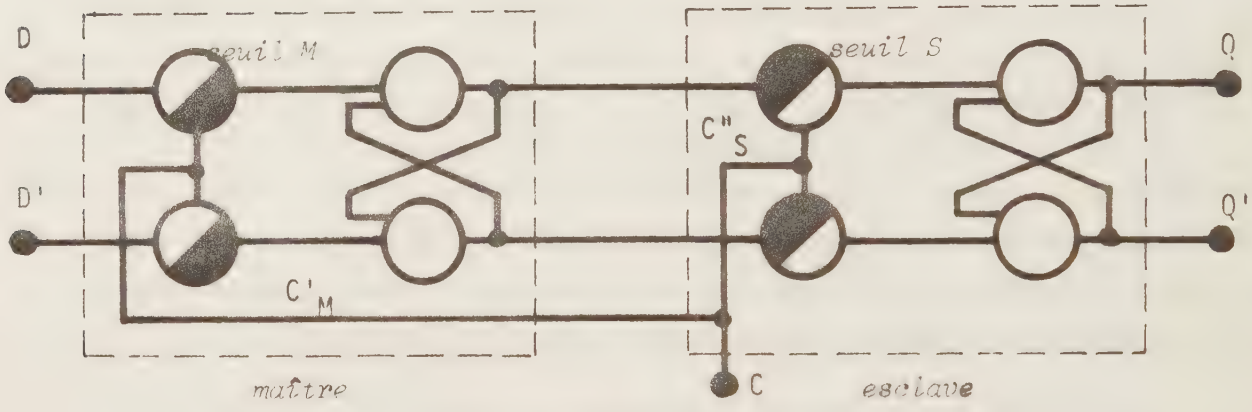
Il ne peut donc être question d'utiliser le même type de porte et de complémenter la commande ; de plus, cette complémentation introduit un retard : la mutuelle exclusion des commandes est alors impossible (cette remarque s'oppose à l'exemple [DIE, fig 4. 17].

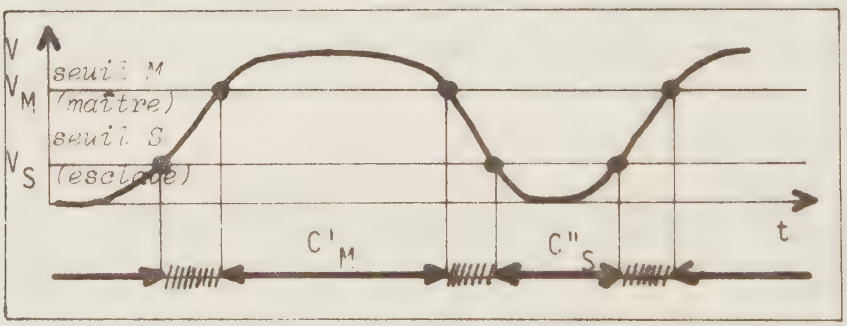
Une bascule maître - esclave ne nécessite pas obligatoirement une double commande. Le principe de fonctionnement est alors fondé sur la remarque suivante :

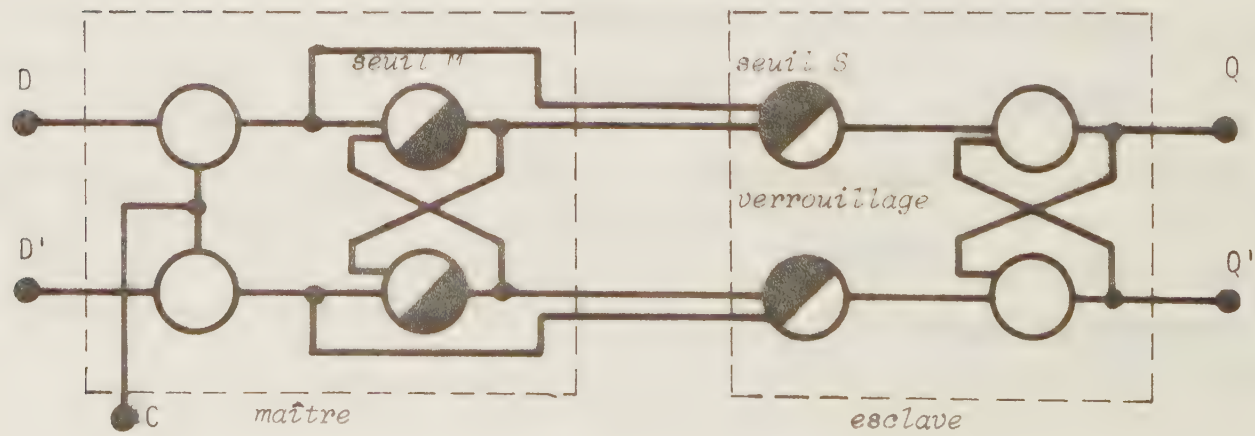
L'établissement d'un état propre de l'esclave est dû aux sorties d'état du maître. Il en résulte qu'un état propre de l'esclave doit être maintenu uniquement lorsque le maître change d'état propre, pendant toute la durée de l'établissement. Comme l'établissement et le maintien d'un état propre d'une cellule binaire de mémoire sont véhiculés par deux signaux d'entrée distincts, il suffit de verrouiller l'entrée de maintien pendant la durée de l'établissement de l'état propre du maître. Le signal de verrouillage est toute désigné : c'est l'entrée d'établissement du maître.

Un même signal sert alors à établir le maître et à verrouiller l'esclave. Il y a donc encore un problème de détection d'un même seuil par deux fonctions extrêmales distinctes. Comme le verrouillage doit débuter avant l'établissement et finir après, les portes d'entrée de l'esclave doivent détecter les frontières des composantes neutres du signal commun, alors que la cellule "RS" du maître doit détecter les frontières des composantes maximales du même signal.

Nous obtenons donc les mêmes contraintes de "double seuil" que pour l'autre type de bascule "maître - esclave". Autrement dit, les conditions de fonctionnement sont identiques pour les deux types de bascule "maître - esclave".







bascules "maître-esclave"

NOTE 1

Nous montrons dans cette note que le fonctionnement logique de la bascule RS ne peut être expliqué sans l'aide du principe des effets retardés.

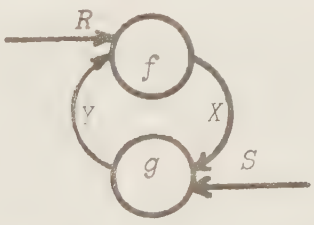
LA BOUCLE DE RETOUR

Hypothèses

Nous postulons au départ que les fonctions combinatoires possèdent uniquement deux variables d'entrée et une variable de sortie. De plus, nous postulons que les alphabets associés à tous les signaux sont identiques et binaires (logiques à seuil unique).

Propriétés logiques

Nous avons vu qu'un alphabet d'entrée ternaire (deux caractères, plus le séparateur) est logiquement nécessaire pour la mémorisation. Comme les alphabets sont binaires, l'"entrée" ne peut être constituée que d'un produit de signaux. De plus, une "boucle de mémorisation" est nécessaire. Dans ces conditions, la structure minimale de l'élément de mémorisation est constituée de deux fonctions combinatoires f et g bouclées l'une à l'autre.



$X \leftarrow f\ (R,Y)$

$Y \leftarrow g\ (S,X)$

L'entrée est le produit des signaux R et S. Par contre il y a plusieurs possibilités pour la sortie :

a) Signal X
b) Signal Y
c) Produit X.Y

Les quatres signaux ont le même alphabet binaire noté {0, 1}. L'alphabet de mémorisation est pour le moment indéterminé mais il possède trois caractères notés {+, - , e}. Comme l'entrée est le produit R.S, chaque caractère possède une composante sur R et une autre sur S. L'alphabet est alors noté :

$$\{r_+ \; s_+ \; , \; r_- \; s_- \; , \; r_e \; s_e \}$$

Le problème logique consiste à déterminer les différentes valeurs possibles de ces composantes, relativement au type des fonctions f et g et de l'alphabet binaire {0, 1}.

Relations de mémorisation

1) changement d'état

L'état de l'élément de mémorisation est caractérisé par celui des signaux X et Y. Les caractères d'entrée $r_+ \; s_+$ et $r_- \; s_-$ sont supposés mettre l'élément dans deux états distincts $E_+ \; (x_+ \; y_+)$ et $E_- \; (x_- \; y_-)$, quelquesoit l'état antécédent. Si nous supposons que les deux composantes d'un caractère d'entrée ont une action symétrique, cette propriété se traduit par les relations suivantes :

$$\forall y, \; x_+ = f \; (r_+, \; y) \qquad (3) \quad \forall y, \; x_- = f \; (r_- , \; y)$$
$$\forall x, \; y_+ = g \; (s_+, \; x) \qquad (4) \quad \forall x, \; y_- = g \; (s_- , \; x)$$

2) mémorisation

L'apparition du séparateur $(r_e, \; s_e)$ après un des caractères $(x_+ \; y_+)$ ou $(x_- \; y_-)$ ne doit pas modifier l'état de l'élément.
Nous avons alors les relations :

$$x_+ = f \; (r_e, \; y_+) \qquad (1) \quad x_- = f \; (r_e, \; y_-)$$
$$y_+ = g \; (s_e, \; x_+) \qquad (2) \quad y_- = g \; (s_e, \; x_-)$$

P1 : Il en résulte que les composantes x_+ (resp. y_+) et x_- (resp. y_-) doivent être distinctes. En effet, d'après la relation 2 $x_+ = x_- \Rightarrow y_+ = y_-$ et réciproquement, d'après la relation 1 : $y_+ = y_- \Rightarrow x_+ = x_-$. Cette double égalité est contradictoire avec le fait que les états $E_+ \; (x_+, \; y_+)$ et $E_- \; (x_-, \; y_-)$ doivent être distincts.

P2 : Il découle de la relation 3 et de la propriété P1 que les composantes r_+ et r_- doivent être distinctes. De même, d'après la relation 4 et la propriété P1, les composantes s_+ et s_- doivent être distinctes.

P3: CONTRADICTION: comme les alphabets sont binaires, la composante r_e est égale à r_+ OU BIEN à r_- (si $r_e = r_+ = r_-$, alors d'après 3: $x_+ = x_-$, ce qui est contradictoire avec P1 ; idem pour $s_e = s_+ = s_-$). Si l'on reporte cette égalité dans la relation 1 , nous obtenons une contradiction avec la relation 3 :

Si $r_e = r_+ \Rightarrow x_+ = f(r_+, y_+)$ et $x_- = f(r_+, y_-)$ alors que d'après 3 :

$\forall y,\ f(r_+, y) = x_+$. Il en résulterait que $x_+ = x_-$!

Si $r_e = r_-$ nous obtenons, d'une part : $x_+ = f(r_-, y_+)$
et d'autre part : $\forall y\ x_- = f(r_-, y)$!

La contradiction provient des relations 3 et 4 qui sont trop strictes. Elles doivent être remplacées par les suivantes :

$$\forall y,\ x_+ = f(r_+, y) \quad (5) \quad x_- = f(r_-, y_-)$$
$$y_+ = g(s_+, x_+) \quad (6) \quad \forall x,\ y_- = g(s_-, x)$$

ou bien :

$$x_+ = f(r_+, y_+) \quad (7) \quad \forall y \quad x_- = f(r_-, y)$$
$$\forall x,\ y_+ = g(s_+, x) \quad (8) \qquad y_- = g(s_-, x)$$

Il en résulte que les deux composantes d'un caractère d'entrée (r_+, s_+) ou (r_-, s_-) n'ont pas une action symétrique : l'une doit "apparaître" avant l'autre. Ainsi, pour les relations 5 et 6 , r_+ précède y_+ et y_- précède r_- ; ce que nous écrivons :

" Si r_+ alors x_+ et puis si s_+ alors y_+ "
" Si s_- alors y_- et puis si r_- alors x_- ".

D'après les nouvelles relations, nous obtenons finalement les relations suivantes

$$(9)\ x_+ \neq x_- \quad \text{et} \quad y_+ \neq y_- ,$$

(10) $r_+ \neq r_-$ (d'après 9 et(5 ou 7)),

$s_+ \neq s_-$ (d'après 9 et(6 ou 8)),

(11) $r_e = r_+$ ou bien $r_e = r_-$,

$s_e = s_+$ ou bien $s_e = s_-$.

La " table de vérité" de la fonction f ou g est alors de la forme suivante :

f		
R \ Y	y_-	y_+
r_-	x_-	x_+
r_+	x_+	x_+
$r_e = r_+$	x_+	x_+
$r_e = r_-$	x_-	x_+

g		
S \ X	x_-	x_+
s_-	y_-	y_-
s_+	y_-	y_+
$s_- = s_e$	y_-	y_-
$s_+ = s_e$	y_-	y_+

Relativement à l'alphabet binaire {0, 1} des signaux, la fonction booléenne remplie par f ou g dépend des conventions choisies 1) pour les composantes de l'alphabet de mémorisation ($r_+ s_+$, $r_- s_-$, $r_e s_e$)

2) pour les états"

$E_+ (x_+, y_+)$ et $E_- (x_- y_-)$

Quelquesoit la convention choisie pour X et Y, le couple de fonction (f, g) est de l'un des trois types suivants (la convention sur les entrées importe peu, car elle peut être modifiée) :

(et, ou) ,

($\overline{ou}$, $\overline{ou}$) (ie : nor, nor),

($\overline{et}$, $\overline{et}$) (ie : nand, nand).

Les deux derniers types vérifient en fait les relations :

$$x = r^{*} + \overline{y} \qquad y = s^{*} + \overline{x}$$

ou bien

$$x = r^{*} \cdot \overline{y} \qquad y = s^{*} \cdot \overline{x}$$

avec $r^{*} = r_{+}$ *ou* r_{-} ; $s^{*} = s_{+}$ *ou* s_{-}

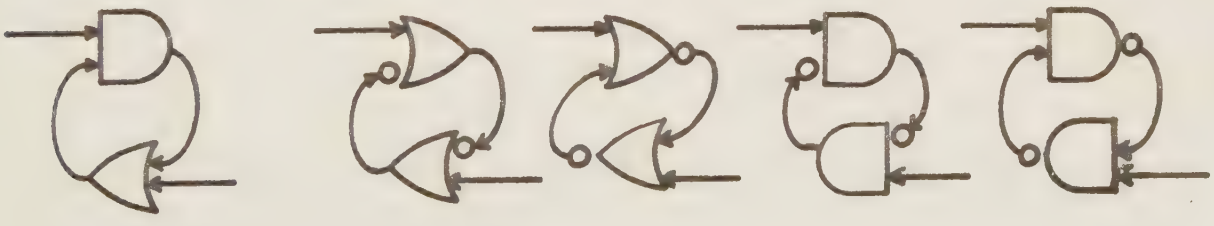
que l'on peut remplacer par:

$$\overline{x} = \overline{r^{*} + \overline{y}} = NOR\ (\ r^{*}\ ,\ \overline{y}\ ) \qquad \overline{y} = \overline{s^{*} + \overline{x}} = NOR\ (\ s^{*}\ ,\ \overline{x}\ )$$

$$\overline{x} = \overline{r^{*} \cdot \overline{y}} = NAND\ (\ r^{*}\ ,\ \overline{y}\ ) \qquad \overline{y} = \overline{s^{*} \cdot \overline{x}} = NAND\ (\ s^{*}\ ,\ \overline{x}\ )$$

Il suffit alors de changer de convention, c'est à dire remplacer l'alphabet des entrées X et Y par son "complémentaire" { 1, 0 }.

Convention	f	g	couple	
$x_{+} = 1 \& y_{+} = 1$	$r + y$ $\overline{r} + y$	$s \cdot x$ $\overline{s} \cdot x$	(et, ou)	même convention pour X et Y
$x_{+} = 0 \& y_{+} = 0$	$\overline{r} \cdot y$ $r \cdot y$	$\overline{s} + x$ $s + x$		
$x_{+} = 1,\ y_{+} = 0$	$r + \overline{y}$ $\overline{r} + \overline{y}$	$s + \overline{x}$ $\overline{s} + \overline{x}$	(nor, nor)	convention opposées pour X et Y
$x_{+} = 0,\ y_{+} = 1$	$r \cdot \overline{y}$ $\overline{r} \cdot \overline{y}$	$s \cdot \overline{x}$ $\overline{s} \cdot \overline{x}$	(nand, nand)	



type 1 *type 2* *type 3*

NOTE 2

LE TORE MAGNETIQUE

Nous venons de définir la fonction combinatoire de mémorisation. Cependant, l'utilisation d'éléments combinatoires n'est pas nécessaire pour obtenir un élément de mémorisation. Il existe en effet des matériaux qui possèdent intrinsèquement cette propriété, en particulier les matériaux magnétiques. Malheureusement, l'emploi de ces matériaux est limité par certaines de leurs propriétés physiques:

- *volume nécessaire "important",*
- *consommation d'énergie,*
- *inertie (qui limite la fréquence de fonctionnement).*

Le tore magnétique est certainement l'élément de mémorisation de ce type le plus connu; c'est pourquoi nous le prenons comme exemple pour mettre en évidence les notions de séparateur et de repérage.

La bascule magnétique

Le tore magnétique peut être considéré comme un élément bistable ayant:

a) une entrée: le signal de mise en un état d'aimantation. La grandeur physique associée au signal d'entrée est la variation de flux magnétique dû à une variation d'intensité du courant d'entrée. L'alphabet élémentaire associé à ce signal comporte trois caractères:

1) *impulsion positive de courant (+),*
2) *impulsion négative de courant (-),*
3) *courant nul (e).*

b) une sortie: le signal de "lecture" dont la grandeur caractéristique est la force contre-électromotrice induite par un changement d'état d'aimantation. Trois types de composantes physiques peuvent être discernées pour ce signal de sortie:

1) *impulsion positive de tension,*
2) *impulsion négative de tension,*
3) *tension de référence (ie: pas d'impulsion).*

L'alphabet associé au signal de sortie comporte trois caractères:

- *deux caractères "utiles" : a) présence, b) absence d'impulsion relativement à la présence des impulsions d'entrée d'un sens donné,*
- *un caractère neutre qui est la tension du signal en dehors de la présence de ces impulsions d'entrée; cette tension n'est pas forcément celle de référence.*

Cet alphabet de sortie est ainsi repéré par rapport au signal d'entrée.

Mémorisation

Une impulsion est recueillie en sortie uniquement lorsque le tore change d'état. Il en résulte que - si l'on connait le sens de l'impulsion d'entrée - on ne peut déduire que la conclusion suivante de l'observation de la présence ou absence de l'impulsion de sortie:

" l'impulsion d'entrée (par rapport à laquelle la sortie est observée) est ou n'est pas la première impulsion qui suit une impulsion d'entrée de sens opposé".

Dans ces conditions, comment "mémoriser" une suite quelconque (indéterminée) de caractères? Nous rappelons la méthode - universellement connue-, pour montrer l'importance du caractère séparateur d'entrée.

ECRITURE

a) Le tore est mis dans un état de référence à l'aide d'une impulsion d'entrée dont le sens est défini par le choix de l'état de référence (pris parmi les deux états stables).

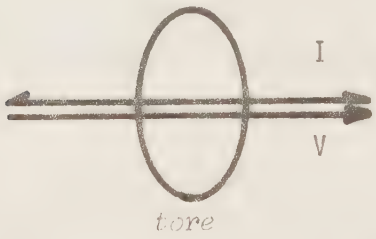
b) L'état du tore est alors "commuté" en fonction de "l'information" à mémoriser: cette information conditionne la présence ou l'absence d'une impulsion de commutation.

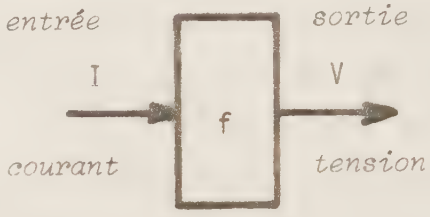
Cette méthode implique qu'il n'y ait aucune impulsion d'entrée entre ces deux impulsions. Celà implique donc la présence du séparateur (courant nul) entre les deux impulsions. L'alphabet d'entrée de mémorisation est alors différent de l'alphabet élémentaire <-, +, e>: $A_E = < - , - e + , e >$.

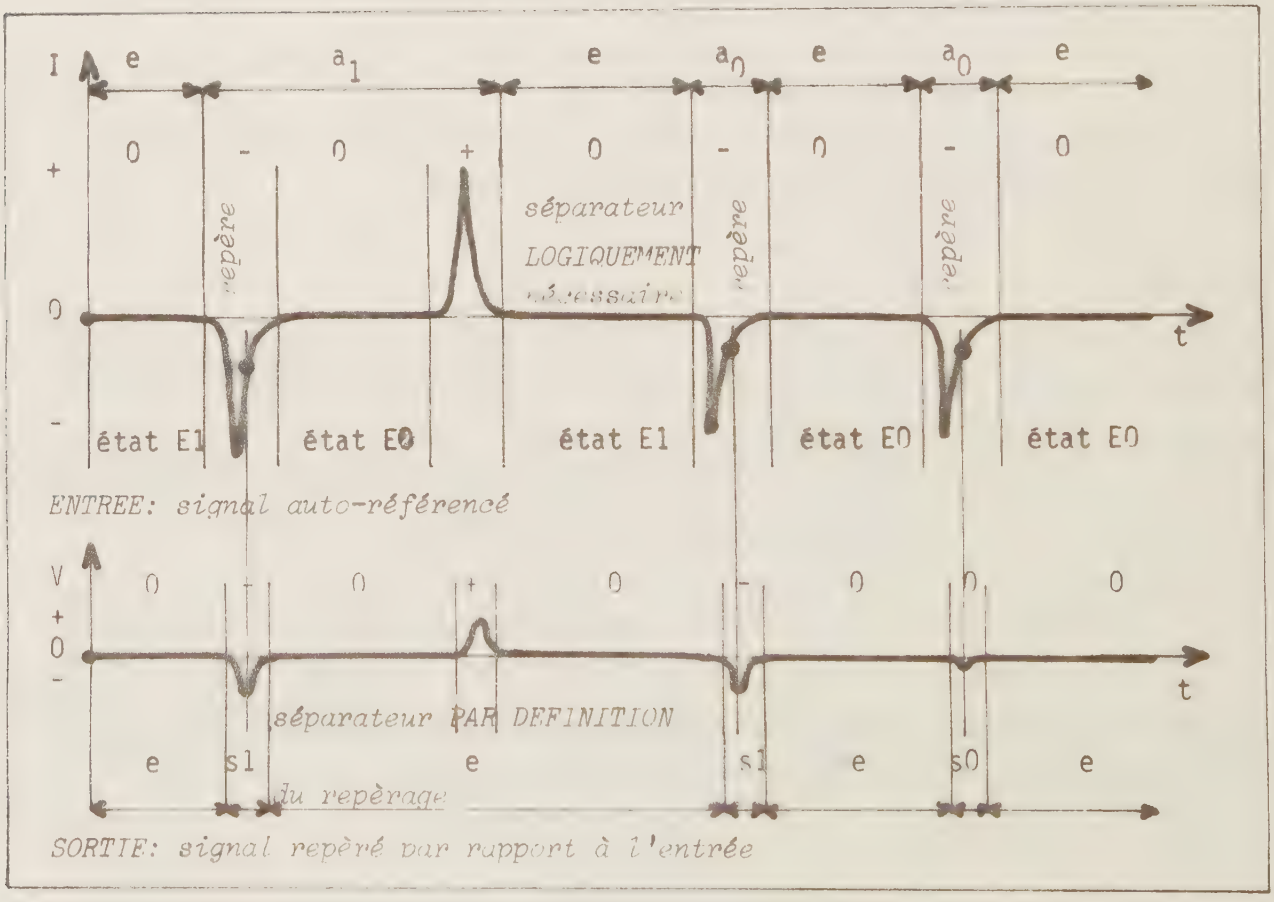
LECTURE

La lecture est plus simple car il suffit d'émettre une impulsion d'entrée de sens déterminé pour connaître l'état du tore (en observant la présence ou l'absence d'une impulsion de sortie qui indique un changement d'état). Cette impulsion d'entrée ne correspond pas à un caractère d'entrée et peut modifier l'état du tore. Habituellement, il est dit que l'état du tore doit être restauré à l'aide d'une impulsion de sens opposé.

.../...







tore magnétique

Il s'agit dans ce cas d'une écriture où le caractère d'entrée correspond au caractère lu. Cette constatation est d'ailleurs exploitée dans certains ordinateurs, pour lesquels la mémoire fonctionne par "demi-cycle": le caractère "restauré" n'est pas le caractère lu. Cette méthode permet d'effectuer une permutation entre un registre "maître-esclave" et la mémoire en un seul cycle.

Pour être valide, l'opération de lecture doit être le plus proche prédécesseur de l'opération d'écriture logiquement associée. Ces deux opérations ne doivent être séparées que par l'état neutre (caractère séparateur) du signal d'entrée.

Logique ternaire

Cette étude du tore magnétique conduit à une conclusion qui nous semble fondamentale:

la notion de mémorisation implique une logique au minimum ternaire.

En effet, si le signal d'entrée ne possède pas de composante séparateur, il est alors uniquement constitué de composantes "logiques" adjacentes. La durée des évènements porteurs d'information est alors égale à l'intervalle de temps qui sépare deux évènements consécutifs: la notion de mémorisation perd ainsi tout son sens. Cette conclusion nous permet de dire que, en logique binaire, les notions de mémorisation et d'inversion se confondent.

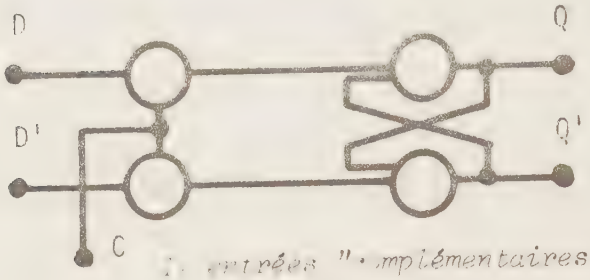
La partition au minimum ternaire de l'ensemble des composantes du signal d'entrée correspond à un découpage physique: au minimum trois types de composantes doivent être physiquement discernables (logique à double seuil). Celà est dû uniquement à la nature déterministe du repère d'entrée, nature qui résulte des implications logiques de la mémorisation.

La partition de l'ensemble des composantes du signal de sortie ne découle pas des mêmes impératifs. En effet, tout notre raisonnement est fondé sur le signal d'entrée qui nous sert"inconsciemment" de repère: celà apparait clairement sur le schéma précédent. Le signal de sortie a trois types de composantes physiques, mais en fait seul l'état du signal, relativement à un seul type de composante du signal d'entrée, est significatif. Les autres parties importent peu et correspondent au séparateur, relativement à un repère de type partiel: l'état du signal peut y être quelconque. La partition ternaire des composantes du signal de sortie est alors purement abstraite.

NOTE 3

RESUME DES CARACTERISTIQUES DES BASCULES STATIQUES

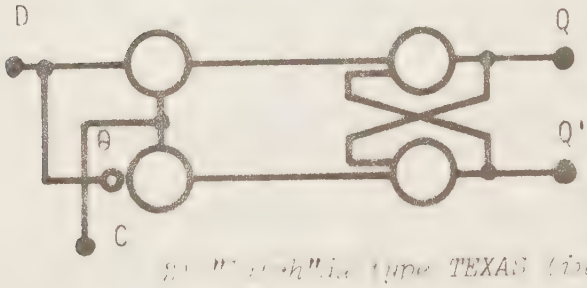
ECHANTILLONNAGE PERMANENT



1) repèreS "mémorisé" de C

2) repère C inclus dans D

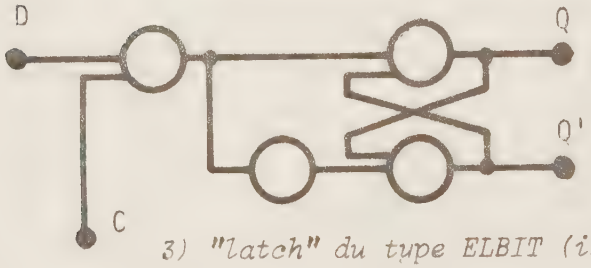
1) entrées "complémentaires" (bascule du type RST)



1) repère S "mémorisé" de C

2) $t_D \leq t_c - \theta$, $T_D \geqslant T$.

2) "latch" du type TEXAS (inversion en aval, ex: SN7475)

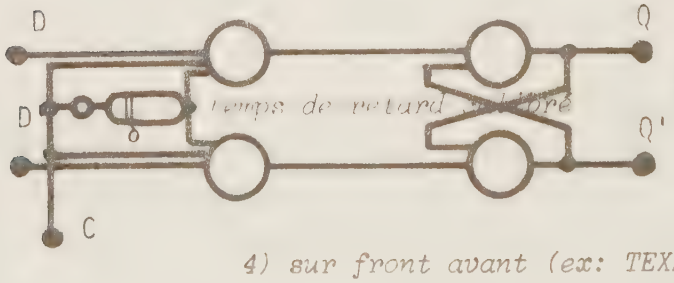


1) repère S "mémorisé" de C

2) repère C inclus dans D

3) "latch" du type ELBIT (inversion en amont)

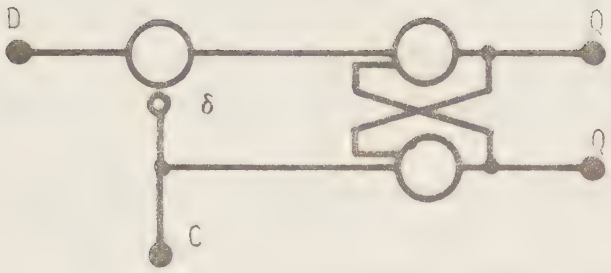
ECHANTILLONNAGE TEMPORAIRE



1) repère S "mémorisé" de C

2) contrainte minimale en entrée:
D stable sur le début de c ($d \geqslant \delta$)

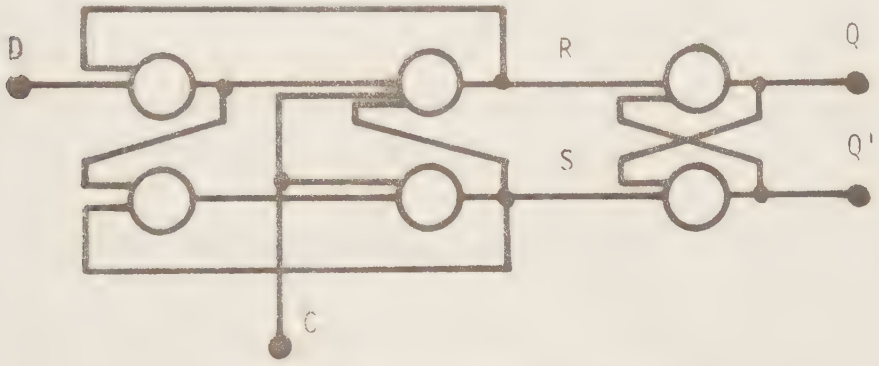
4) sur front avant (ex: TEXAS SN 74111)



1) les relations entre les repères S,C,D dépendent des temps de retard
2) "aléa " à l'ouverture
3) temps de retard δ calibré

5) sur front arrière (latch du type IBM 360/30)

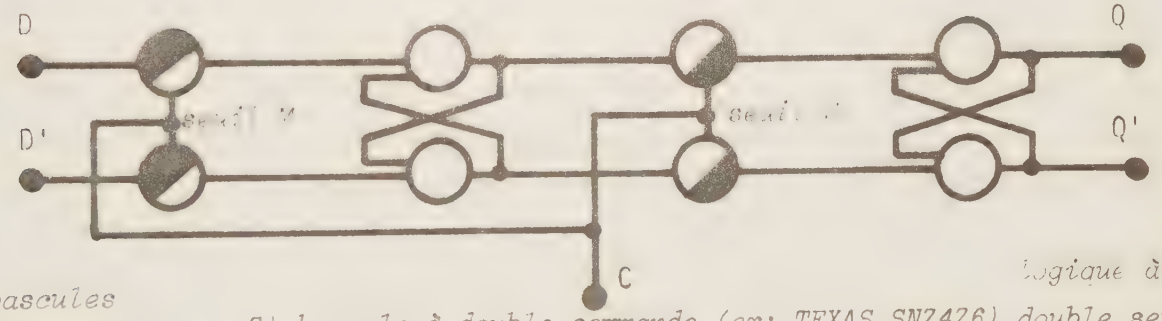
ECHANTILLONNAGE A MEMOIRE



mutuelle exclusion D / (R,S)

6) bascule du type D-TEXAS (SN 7474)

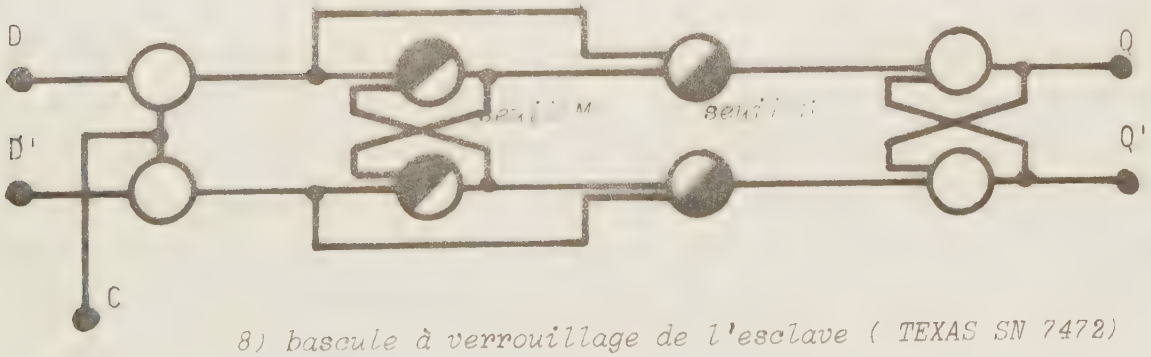
BASCULES "MAITRE - ESCLAVE"



bascules "[illegible]"

logique à

7) bascule à double commande (ex: TEXAS SN7476) double seuil



8) bascule à verrouillage de l'esclave (TEXAS SN 7472)

Contraintes temporelles

Quelquesoit le type de la bascule, les contraintes temporelles sont semblables :

1 - La durée de la commande C doit être supérieure au temps de "bouclage" d'une cellule "RS", c'est à dire deux fois un retard moyen θ_m : $d_C > 2\,\theta_m$

2 - Le signal de donnée doit être stable pendant une durée semblable : $d_D > 2\theta_m$

Notion d'aléa

Pour que les éléments physiques assument leur fonction logique, il faut
- d'une part, que la durée et la position mutuelle des signaux respectent certaines contraintes temporelles,
- d'autre part, que les temps de retard des éléments vérifient certaines contraintes, du moins pour certaines bascules.

Autrement dit, l'ensemble des signaux doit être cohérent (ie : "synchrone") : si les contraintes ne sont pas respectées, le fonctionnement logique n'est pas garanti. Vu sous cet angle, la notion d'aléa doit être remplacée par celle de contrainte temporelle.

Nous avons vu, dans deux cas (latch "Elbit" et latch "IBM"), qu'il pouvait y avoir des "composantes parasites". Comme elles précèdent les composantes effectives d'établissement, ces composantes n'influent pas sur l'effet final. De plus, elles ont lieu pendant l'échantillonnage. Comme la sortie n'est pas observable - par définition logique - à cet instant, leur effet temporaire importe peu. Ces composantes parasites ne sont aléatoires que dans la mesure où leur apparition dépend de l'état du signal d'entrée.

Ces composantes parasites ont été décelées pour des bascules dissymétriques. Cette dissymétrie - cause des "aléas" - est en fait la raison d'être de ces bascules, car elle permet de supprimer un élément logique, donc de réduire le nombre de composants physiques. Corriger cet aléa par adjonction d'un élément supplémentaire n'a alors aucun sens. De plus, la correction est toute indiquée : il suffit de choisir une des autres bascules décrites qui n'a pas ce défaut !

En résumé, la terminologie d'"aléa " nous semble mal appropriée car toutes les bascules décrites remplissent leur fonction logique, si les contraintes énoncées sont vérifiées ; leur fonctionnement n'est donc pas aléatoire.

Quelquesoit l'élément physique, il y a toujours des contraintes d'emploi (ie : un mode d'emploi)!

CHAPITRE IV

Ce chapitre est consacré aux propriétés spécifiques des systèmes séquentiels qui possèdent leur propre référentiel de temps. Cette particularité implique d'une part que leur synthèse est effectuée en termes de transfert et d'autre part que leur analyse temporelle est rapportée à leur propre référentiel de temps et n'est plus relative à leurs entrées / sorties. Enfin, ces systèmes sont classés en fonction de leur mécanisme de synchronisation avec d'autres systèmes du même type.

APPLICATION DU MODELE
AUX SYSTEMES A TEMPS PROPRE:

ETUDE DES HORLOGES

I - Cycle de commandes
II - Diagramme des temps
III - Domaines temporels
IV - Structure d'une horloge

LES HORLOGES

I -CYCLE DE COMMANDES

1.1. Notion de transfert élémentaire

Dans la théorie des machines séquentielles, un système séquentiel est défini de la façon suivante :

1) Repère temporel : la référence de temps est unique et dénombrable ; l'échelle de mesure du temps est donc une suite de dates

notée t_o, t_1, ... , t_n, t_{n+1}, ...

2) Etat : l'état du système est défini - pour tout point t_n de la référence de temps - par l'état de tous les signaux du système. Globalement, un système séquentiel peut être décomposé sur trois types de signaux :

- entrée I,
- sortie O,
- "état interne" S.

Le fonctionnement du système est alors caractérisé par deux fonctions f et g :

1) fonction d'état f : $S(t_{n+1}) = f(S(t_n), I(t_n))$

2) fonction de sortie g : $O(t_n) = g(S(t_n))$
(Machine de Moore)

$O(t_n) = g(S(t_n), I(t_n))$
(Machine de Mealy)

L'étude des systèmes séquentiels est fondée sur la décomposition du système en un ensemble d'éléments séquentiels de base : ie : les cellules de mémoire $M_{i=1,p}$. Ces cellules sont supposées opaques. Les relations précédentes sont alors remplacées par le système de relations suivant :

$$[\,i\,]\quad \forall i \mid i = 1,p : \quad M_i\,(t_{n+1}) = f\,(I(t_n),\, M_1(t_n),..M_i(t_n),..,M_p\,(t_n)),$$

$$O(t_n) = g(I(t_n)^{*},\, M_1(t_n),...,M_i(t_n),..,\, M_p(t_n))$$

pour une "machine de Mealy"(*) où $M_i\,(t_{n+1})$ représente l'état (propre) de la cellule de mémoire à la date t_{n+1}.

La relation [i] caractérise un transfert élémentaire ; elle correspond à la relation de transfert que nous avons introduite dans le premier chapitre :

l'état $M_i(t_{n+1})$ est l'effet ,

dont les états $I(t_n)$, $M_1(t_n)$,.., $M_i(t_n)$,..,$M_p(t_n)$ sont la cause.

Un transfert élémentaire est effectif lorsque les états propres $M_i(t_{n+1})$ et $M_i(t_n)$ sont différents.

Puisque la référence de temps est unique pour toutes les cellules de mémoires M_i, le nombre de transferts élémentaires par période de l'horloge (ie : de la référence de temps) est égal au nombre de cellules de mémoire.

La notion de transfert élémentaire est ainsi strictement liée à l'ensemble des cellules de mémoire.

Le choix d'une référence de temps unique a alors un inconvénient majeur : la définition d'un seul transfert effectif par unité de temps implique la définition de p-1 transferts "neutres". Celà nécessite physiquement :

-1) p fonctions de porte logique dont p-1 servent à "inhiber" la référence de temps

-2) et que toutes les cellules de mémoire soient opaques car leur état est constamment redéfini en fonction de celui des autres.

Cet inconvénient peut être supprimé si à chaque cellule de mémoire est associée une référence de temps propre ; où chacune de ces références repère les transferts effectifs dont la cellule associée est le "récepteur".

L'horloge est alors constituée d'un ensemble de références de temps qui sont liées entre elles, afin que l'ensemble des transferts élémentaires soit cohérent.

La période de l'horloge est alors le plus petit commun multiple (PPCM) des périodes de ces différentes références de temps. Une horloge périodique de ce type est un des principaux fondements d'une machine "microprogrammée".

1.2. Synthèse d'un cycle de commandes

1.2.1. PRINCIPE

Le cycle de commandes d'un système séquentiel détermine l'ensemble des transfert élémentaires exécutables pendant une période de l'horloge du système.

La synthèse de ce cycle a pour but de déterminer le nombre minimal de commandes (ie : références) temporelles qui permet de contrôler un ensemble de transferts donné.
Une telle synthèse implique que le système séquentiel soit défini au préalable sous la forme d'un ensemble de cellules de mémoire et d'un ensemble de liaisons entre ces cellules.

L'évaluation du nombre de commandes part du principe que leur durée ou bien l'intervalle de temps qui peut séparer deux commandes peut être quelconque.
Il en résulte que les temps de retard de propagation des signaux à travers les circuits combinatoires n'interviennent pas logiquement dans cette évaluation. Seul le type - transparent ou opaque - des cellules de mémoire importe.

La synthèse du cycle de commandes nécessite donc, en plus de la définition de l'ensemble des transferts élémentaires, la spécification des contraintes imposées sur le type des cellules de mémoire.

Lorsque le type des éléments de mémoire n'est pas spécifié, il est alors supposé être transparent car la structure de ce type de cellule de mémoire est la plus simple. Si l'étude montre qu'une de ces cellules de mémoire est à la fois source et récepteur d'un même transfert, son type est alors remplacé par celui d'opaque.

Les sources et récepteurs d'un transfert extérieurs à la structure (ie : "les entrées - sorties) posent un problème spécifique, celui du découplage temporel. Nous aborderons ce problème par la suite. Il s'avère en fait que le découplage temporel ne peut être réalisé qu'à l'aide d'éléments de mémoire. Une "entrée-sortie" doit donc être considérée comme un élément de mémoire dont on ne peut imposer la propriété d'"opacité ", puisque cet élément sert d'interface. Pour l'étude du cycle de commandes, chaque entrée ou sortie est considérée individuellement comme une cellule de mémoire. Pour une entrée, la commande détermine les contraintes de stabilité alors que pour une sortie, la commande est sa référence de temps qui délimite les intervalles de temps pendant lesquels la sortie est stable.

1.2.2. TERMINOLOGIE

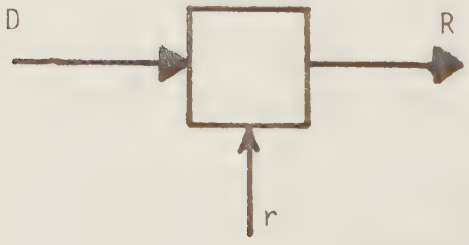
La synthèse d'un cycle de commandes s'exprime en termes de repères dont les composantes définies délimitent les intervalles de temps pendant lesquels l'état d'un signal est stable.

Afin de décrire fonctionnellement un ensemble de transferts élémentaires, nous introduisons un certain nombre de conventions.

A) CELLULE DE MEMOIRE

Une cellule de mémoire peut être caractérisé par trois repères :

- entrée de données : D,
- entrée de commande (ie : référence de temps) : r,
- sortie d'état R.



Un transfert élémentaire dont la cellule de mémoire R est le récepteur est caractérisé par la relation de transfert

$$D \,.\, r \rightarrow R$$

Ce transfert est noté $D \rightarrow R$ et r est supposée être la commande de transfert.

Une cellule de mémoire transparente vérifie les deux relations suivantes :

- 1) "contrainte d'inclusion" : $r \subset D$,
- 2) "mémorisation transparente" : $r \,.\, R = \emptyset$.

Par contre, les repères d'une cellule opaque vérifient les relations suivantes :

- 1) inclusion : $r \subset D$,
- 2) opacité : $r.\, R = r$.

B) FONCTIONS COMBINATOIRES

Trois types fonctionnels sont retenus pour l'étude des transferts.

- 1) La coïncidence : elles correspond à l'intersection des repères :

$$D = D_1 \bullet\bullet\bullet D_i \bullet\bullet\bullet D_n \qquad \begin{matrix} D_1 \rightarrow \\ D_i \rightarrow \\ D_n \rightarrow \end{matrix} \Big| \rightarrow D$$

- 2) La projection : elle caractérise la décomposition d'un repère associé à un alphabet A de dimension N quelconque en N repères binaires mutuellement exclusifs dont les composantes définies correspondent aux différentes occurences d'un même caractère de l'alphabet A.

Si $D_{/N} \rightarrow D_1 |..| D_i |..| D_n$

alors 1) $D_{/N} = D_1 + .. D_i .. + D_n$

2) $\forall i, j \in (1,N) \mid i \neq j : D_i . D_j = \emptyset$

$$D \rightarrow \Big| \begin{matrix} \rightarrow D_1 \\ \rightarrow D_i \\ \rightarrow \\ \rightarrow D_n \end{matrix}$$

1/N

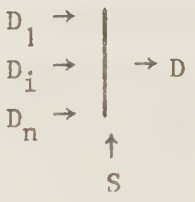
(décodage 1/N)

3) La sélection : elle correspond à l'union (mutuellement exclusive) de repères qui sont l'intersection d'un repère "de donnée D_j" avec une des projections S_i d'un repère de sélection S :

si $D := (D_1 | \ldots D_i | \ | D_N) / S$

alors $S_{/N} \rightarrow S_1 | \ | S_i | \ | S_n$

et $D = D_1.S_1 + \ldots D_i.S_i + \ldots D_n.S_n$



C) COMPOSITION DES TRANSFERTS

Quatre relations sont introduites afin de décrire un ensemble de transferts élémentaires.

- 1) Séquence (notée ,) : cette relation permet d'ordonner les commandes de transfert :

Soient $U \rightarrow X$ et $V \rightarrow Y$ deux transferts élémentaires tels que $y \subseteq X$, alors :

x, y ou $U \rightarrow X$, $V \rightarrow Y$ définit l'ordre des commandes.

- 2) Alternance (notée |) : cette relation caractérise une sélection des commandes de transfert.

Ainsi, l'alternance $(U \rightarrow X | \ V \rightarrow Y)/S$ est équivalente à l'ensemble des relations suivantes :

1) $U.x \rightarrow X$
2) $V.y \rightarrow Y$
3) $S_{/2} \rightarrow S_x | S_y$
4) $x = C . S_x$
5) $y = C.S_y$

- 3) Parallélisme (notée .) : cette relation permet de décomposer un transfert en un produit de transferts indépendants.

Ainsi $(U \rightarrow X).(V \rightarrow Y)$ implique les relations :

1) $x = y$
2) $U.V \rightarrow X.Y$

- 4) Spécification (notée : =) cette relation permet d'abréger l'écriture. Ainsi (X:= expression y) spécifie que chaque occurence du symbole X doit être remplacée par l'expression y.

1.2.3. METHODE

A partir de la carte de la machine - c'est à dire de la définition de l'ensemble des cellules de mémoire et des liaisons entre ces cellules - les "chemins de transfert", entre les différentes cellules, sont déterminés par regroupement des fonctions combinatoires. Ces chemins définissent les sources et les récepteurs d'un transfert, sans tenir compte des fonctions combinatoires traversées.

Le cycle de transferts élémentaires est alors défini à partir de l'ensemble des "chemins de transferts". Cette définition se décompose en deux parties :

- 1) énoncé,
- 2) vérification des contraintes dûes à la transparence des cellules de mémoire.

La non vérification des contraintes entraîne l'introduction de contraintes supplémentaires sur le type des cellules. Ainsi, une cellule qui est à la fois source et récepteur d'un transfert doit être opaque. Si cette cellule est transparente par hypothèse, elle doit être dédoublée (ie : cellule "à double - pont"). Ce dédoublement peut nécessiter une modification de l'énoncé lorsque des clauses de parallélisme sont définies.

Cette règle de dédoublement résulte d'un impératif logique : elle doit être respectée pour que l'"opacité" des transferts soit indépendante des temps de retard. Ce n'est pas un impératif physique, car l'opacité peut être obtenue en "jouant" sur la valeur des temps de retard. Cette possibilité nécessite, dans ce cas, un calcul des temps de propagation qui tienne compte des incertitudes ; l'algorithme de ce calcul est décrit par la suite (cf : "diagramme des temps").

Lorsque les contraintes d'"opacité" ont été déterminées, le cycle de commandes se présente comme un graphe cyclique dont les noeuds sont les commandes et dont les arcs sont valués par les relations de séquence, d'alternance ou de parallélisme.

Les graphes de ce type peuvent être classés en deux catégories :

- 1) les graphes réductibles ,
- 2) les graphes irréductibles.

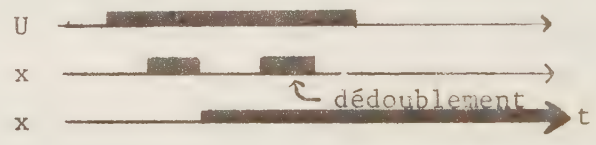
Un graphe est réductible dans la mesure où la durée des commandes et les intervalles de temps qui séparent les commandes (ie : intervalles de position) ne sont pas majorés (ils sont obligatoirement minorés). Il est alors possible de définir une durée ou un intervalle de position commun à plusieurs commandes - ou séquences de commandes - qui peuvent alors être superposées temporellement.

Cette superposition permet de réduire le nombre de références temporelles ; mais elle ne réduit pas le nombre de commandes logiques, qui est posé a priori par la définition de l'ensemble des transferts élémentaires. Seule une re-définition de cet ensemble permet de réduire le nombre de commandes logiques.

Dans le cas d'alternance ou de parallélisme de séquences, la superposition nécessite parfois le dédoublement de la commande d'une même cellule.

Une commande peut être dédoublée sur l'intervalle de temps pendant lequel son "contexte d'entrée" est stable.

Ainsi le transfert $U.x \rightarrow X$ peut être dédoublé en $U.(x \hat{} x) \rightarrow X$:



Pour des cellules de mémoire transparentes, deux commandes de transfert peuvent être superposées si l'état final de l'une est indépendant du contexte initial de l'autre :

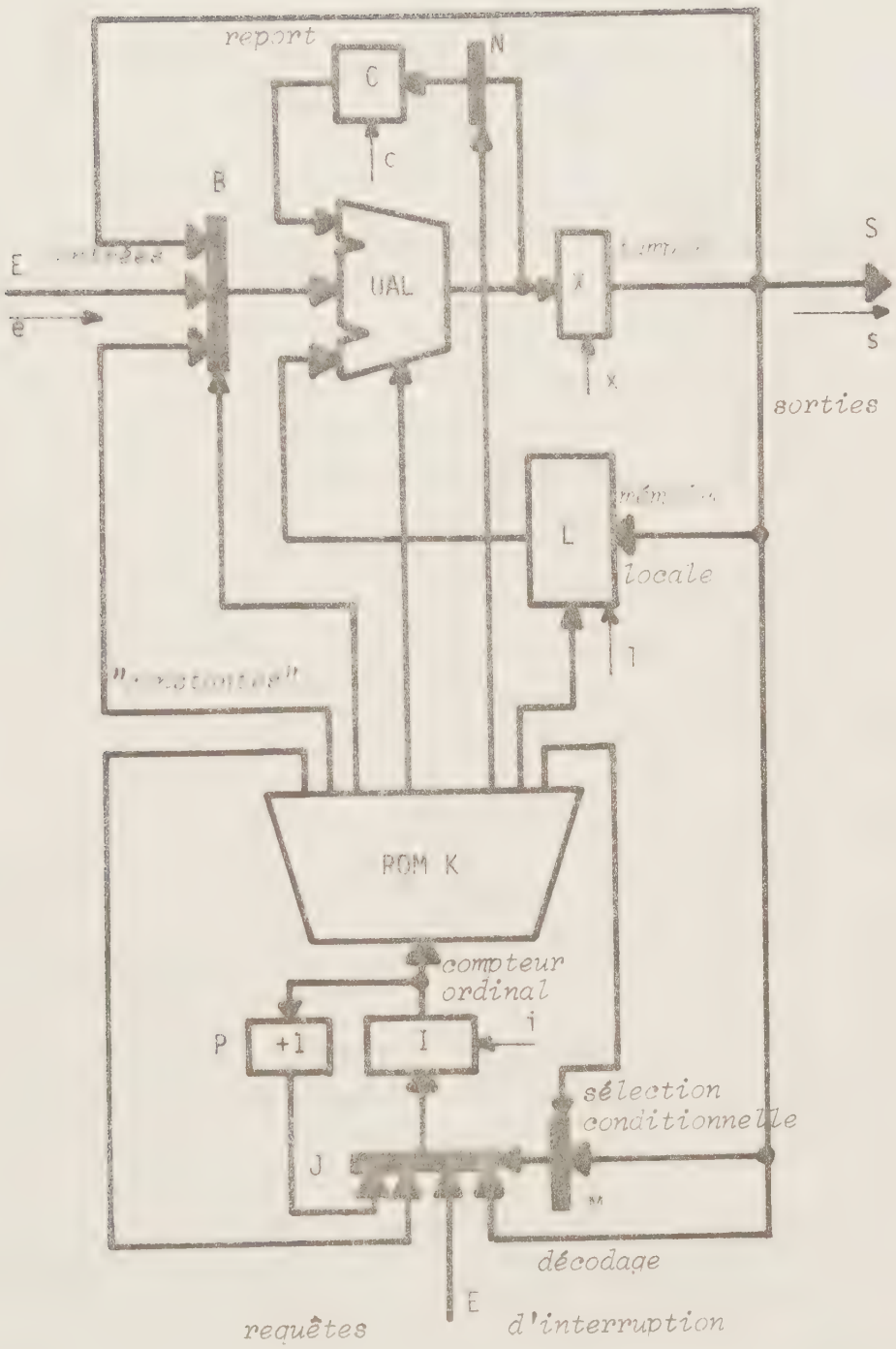
Ainsi les deux transferts $U \longrightarrow X$ et $V \longrightarrow Y$ peuvent être commandés simultanément si

$$X.V = U.Y = \emptyset.$$

L'exemple qui suit illustre la méthode décrite.

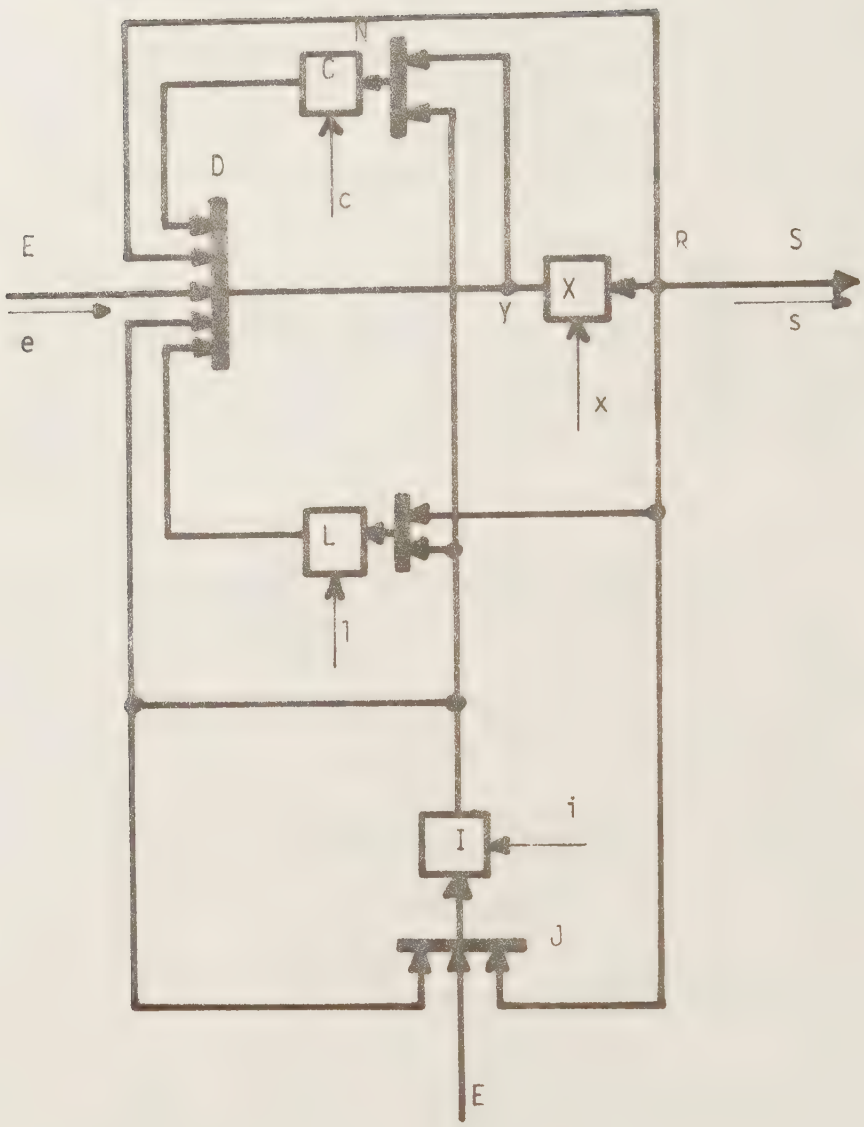
1.3. Exemple

La "carte de la machine" est constituée d'une liaison "séquentielle" entre les entrées E et les sorties S . Cette liaison converge sur un circuit combinatoire : la "boite à opérations" ALU, tamponné par un registre transpa-



CARTE DE LA MACHINE

(" data flow ")



D := ((C|Ø).(L|Ø).(X|E|I|Ø))/I

Y := (X.(C|Ø))/I

R := (L|S|Ø)/I

J := (X|I|E)/I

chemins de transfert

rent X. La sortie de la boite à opérations est "bouclée" à son entrée à travers la cellule de mémoire C de report. La sortie de la liaison (x) est "réinjectée" à son entrée - 1) directement, -2) à travers la mémoire locale L. Le type de cette mémoire est imposé: c'est un élément de mémoire transparent.

La liaison est commandée à partir du circuit combinatoire ROM dont l'entrée est connectée à la sortie du registre d'état I. L'entrée de ce registre a différentes provenances possibles :

- sortie de la liaison (décodage),
- entrée E (requêtes d'interruption),
- sortie incrémentée du registre d'état,
- sortie du circuit de commande ROM.

Deux types de transferts sont envisagés :

- 1) transferts directs "à travers" le registre tampon X,
- 2) permutations entre
  - a) le registre X et
  - b) les autres éléments de mémoire ou les entrées E ou les sorties S.

Le choix du type de transferts est déterminé par la sortie du circuit de commande ROM.

Ensemble des transferts élémentaires : T

$T_0 := J \rightarrow I$

$T_1 := D \rightarrow Y$ (commande $y = x \,.\, (c/I)$)

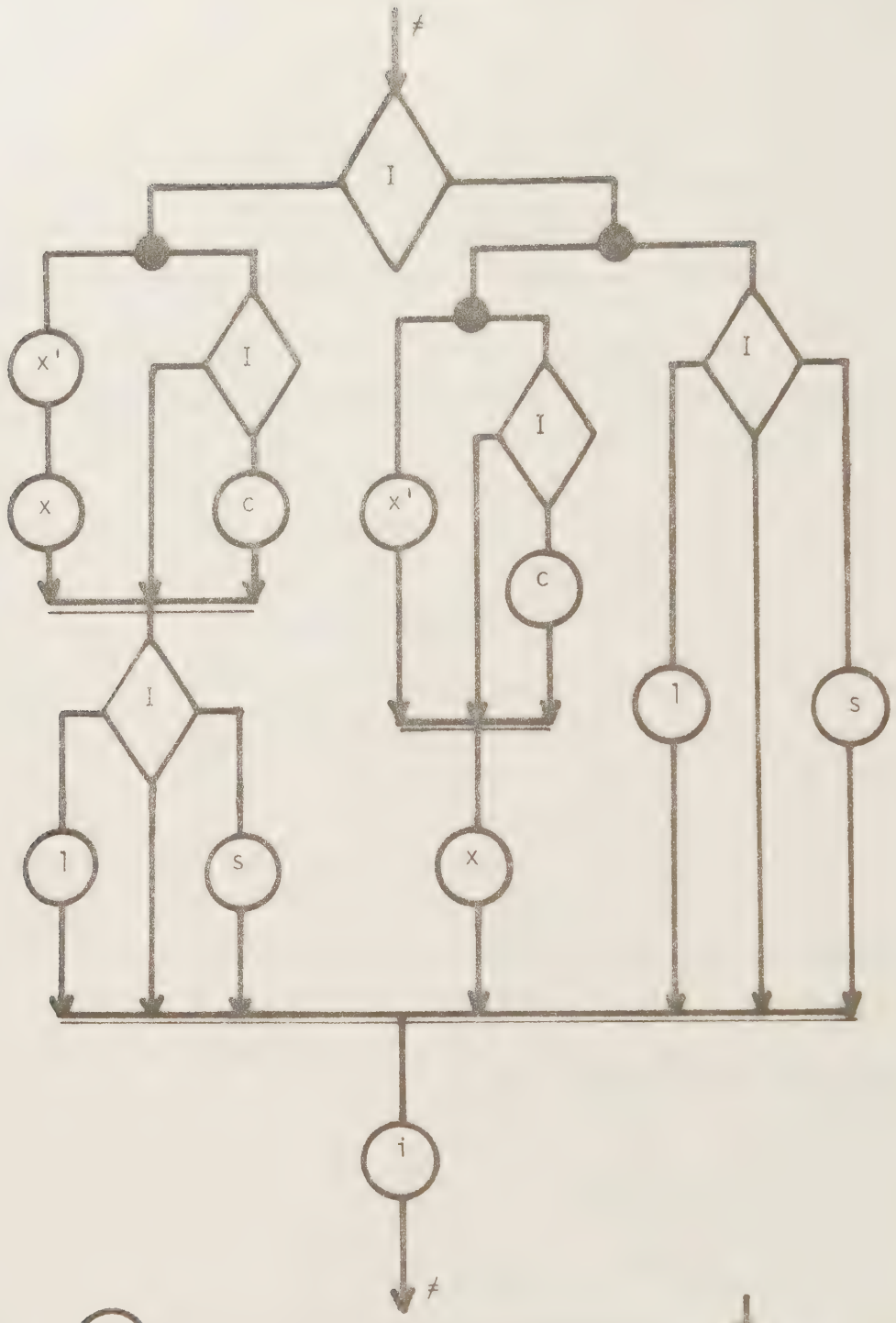
$T_2 := X \rightarrow R$ (commande $r = (1|s)I$)

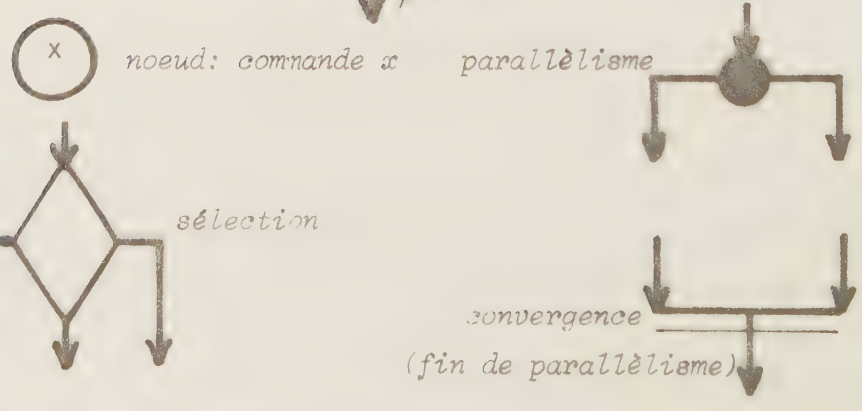
$T := ((T_1, T_2) \,|\, (T_1 \,.\, T_2))/I \,,\, To$

Transfert $To := J \rightarrow I$: I est inclus dans J, le registre I est à la fois source et destination du transfert ; il doit donc être opaque.

Transfert $T_1 := D \rightarrow Y$: X et C sont inclus dans D ; ils doivent donc être opaques. Comme le registre X est par hypothèse transparent, il doit être dédoublé. Si X' est la doublure (transparente) de X, le transfert $D \rightarrow Y$ doit alors être remplacé par :

$T'_1 := (D \rightarrow X', X' \rightarrow X).\ D \rightarrow C$





GRAPHE DES COMMANDES

La clause de parallélisme entraîne un problème de superposition de la séquence de commandes (x',x) avec la commande c ; celle-ci peut être superposée ou bien à x' ou bien à x. La suite de l'analyse définira le choix final.

Transfert $T_2 := X \rightarrow R$: X et R n'ont pas d'élément commun, il n'y a donc pas de problème.

Transfert T_1, T_2 : La commande c ne peut succéder à la commande x qui modifie le contexte initial du transfert $D \rightarrow C$ (X est inclus dans D)
La séquence de commandes est alors :

$$(x', x).c,r$$

Transfert $T_1 . T_2$: La clause de parallélisme entraîne un problème de superposition des commandes r, (x',x) et c. L'élément transparent X est à la fois récepteur de la séquence $D \rightarrow X'$, $X' \rightarrow X$ et source du transfert $X \rightarrow R$. Cela implique que le transfert $X \rightarrow R$ précède le transfert $X' \rightarrow X$. D'autre part, R et D ont un élément transparent commun L qui apparaît comme source du transfert $D \rightarrow X'$ et récepteur du transfert $X \rightarrow R$. Cela implique que le transfert $D \rightarrow X'$ précède le transfert $X \rightarrow R$. Enfin, la commande c ne peut succéder à la commande r qui modifie le contexte initial du transfert $D \rightarrow C$.
La séquence de commandes du transfert $T_1 . T_2$ est finalement :

$$((x',r).c),x$$

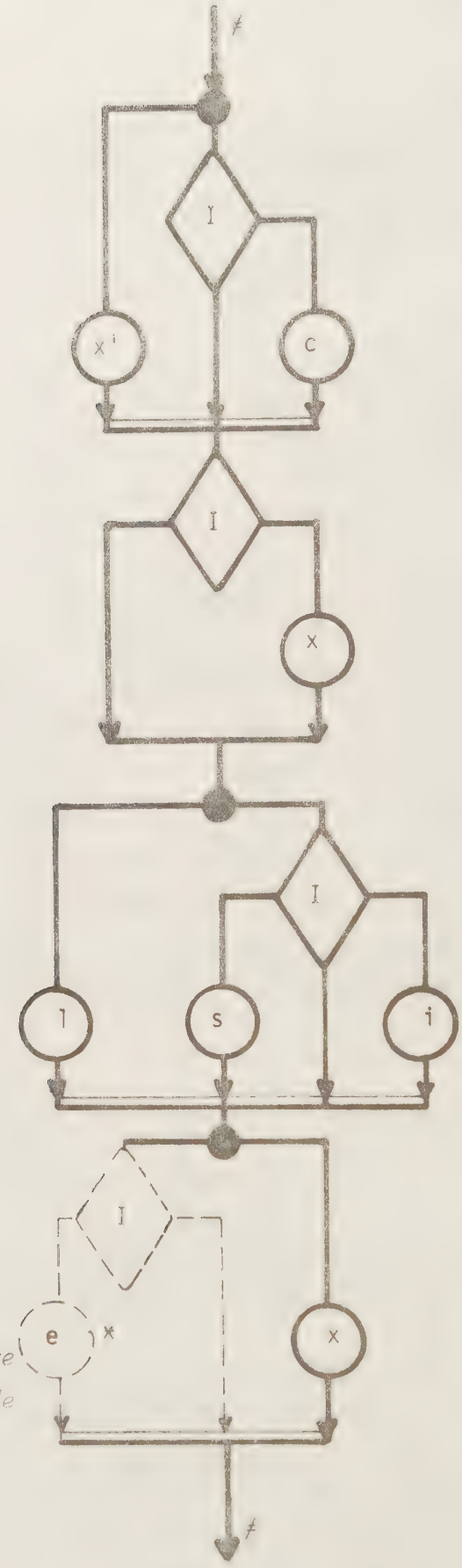
Cycle de commandes T : ce cycle a ainsi pour expression :

$$((((x',x).c),x)|(((x',r).c),x))/I,i$$

1) La commande c doit être superposée à la commande systématique x' (ie : cette commande x' apparaît dans tous les cas).

2) La commande x peut être dédoublée avant ou après la commande r car celle-ci ne modifie pas son contexte d'entrée X'. La seconde occurrence de x (après la commande r) est alors systématique. L'expression du cycle de commandes est alors : (x'.c,x/I,r,x),i. Comme la deuxième occurence de la commande x est systématique, elle ne dépend pas du contexte I et peut être permutée avec la commande i, laquelle peut alors être superposée avec la commande r. Un cycle minimal de commandes est ainsi obtenu :

$$(x'.c,x/I,r.i,x)$$



phases de l'horloge

P3

P4

P1

P2

commande d'échantillonnage (cf. structure de l'horloge)

GRAPHE REDUIT

II - DIAGRAMME DES TEMPS

Le cycle de commandes a été déterminé à partir de la définition de la carte de la machine et d'un ensemble de transferts élémentaires. Seuls le nombre et l'ordre des commandes temporelles ont été déterminés. Il reste maintenant, pour spécifier complètement les caractéristiques de l'horloge, à déterminer les contraintes temporelles

- 1) sur la durée des commandes,
- 2) sur les intervalles de temps qui séparent ces commandes.

Les contraintes temporelles résultent - d'une part des temps de retard introduits par les éléments combinatoires - et d'autre part des caractéristiques temporelles de chaque élément de mémoire. Ces contraintes ne sont pas forcément un inconvénient ; ainsi, lorsque les temps de retard ne sont pas négligeables, une cellule de mémoire transparente peut faire fonction de cellule opaque.

L'évaluation des contraintes permet de définir les valeurs minimale et maximale de la période du cycle de commandes.
Cette évaluation nécessite

- 1) la définition des caractéristiques temporelles des éléments combinatoires et séquentiels du système,
- 2) la définition de l'ensemble (ie : le cycle) des commandes de transferts.

2.1. Repérage et mesure des caracteristiques temporelles

Nous avons vu que deux composantes de signal adjacentes, dont la valeur des grandeurs caractéristiques est identique, sont indiscernables. Il en résulte que seuls les "points" de transition, entre deux composantes distinctes, sont observables. Si les dates de transition sont connues, la durée des composantes peut être évaluée. Cependant, comme la durée n'est pas

un invariant de la fonction de retard, la notion de point n'a pas de sens à l'intérieur d'une composante. Les seuls "points" "observables" sont donc les points de transition.

Repérage relatif

Une date de transition (effet) ne peut être repérée que par rapport à celle de la transition qui en est la cause (ou vice-versa). De proche en proche, il est ainsi possible de lier les dates de transition entre elles. Cependant, l'affiliation de ces points permet, au plus, de repérer un seul point par signal. Il reste alors à "repérer" entre eux deux points de transition d'un même signal. Pour la plupart des signaux (P.ex : les signaux de "données"), ce repérage est impossible. Il faut alors "remonter l'arbre généalogique" de chaque signal et déterminer s'il n'est pas issu d'un signal pour lequel ce repèrage est possible. Si il en est ainsi, le signal d'origine est obligatoirement un signal d'horloge; dans le cas contraire, le signal d'origine est un signal d'entrée. Il reste encore à définir le repèrage mutuel des points de transition d'un tel signal d'horloge. Le principe est le même que pour le repérage ponctuel entre deux signaux : il faut que chaque point de transition d'un signal d'horloge soit le transformé de son prédécesseur. Les caractéristiques temporelles de la transformation déterminent alors la position relative de deux points successifs.

Par affiliation, il est ainsi possible de comparer entre elles les dates de deux points quelconques. Cependant, ce type de comparaison est limité par le cumul des incertitudes sur la valeur des caractéristiques temporelles des éléments. Ce type de comparaison ne peut donc être effectué qu'à une échelle de temps "locale". Il n'y a que deux moyens de"s'en échapper" : il faut définir tous les évènements - caractérisés par des points de transition-

1) ou bien à l'aide de signaux périodiques;

2) ou bien à l'aide de signaux mutuellement exclusifs : une échelle des temps ne peut dans ce cas être définie, mais l'indépendance temporelle des évènements est alors garantie.

Mesure des caratéristiques temporelles des éléments

Le système de repérage décrit suppose que les caractéristiques temporelles des transformations successives d'un point de transition soient connues. Dans la pratique actuelle, ces caractéristiques sont mesurées et non calculées d'après un modèle. La mesure est effectuée par comparaison d'un point de transition du signal d'entrée avec son "transformé" du signal de sortie, relativement à un référentiel absolu, c'est à dire qui est le même pour toutes les mesures. Ce référentiel est "synchronisé" sur un des deux points de transition.

A cause de la fabrication en grande série, les valeurs caractéristiques sont majorées. A l'intérieur d'un "lot" de fabrication, la "dispersion" sur ces valeurs - mesurées pour chaque élément - est inférieur aux bornes majorantes garanties.

2.2. Domaine temporel arborescent d'une commande.

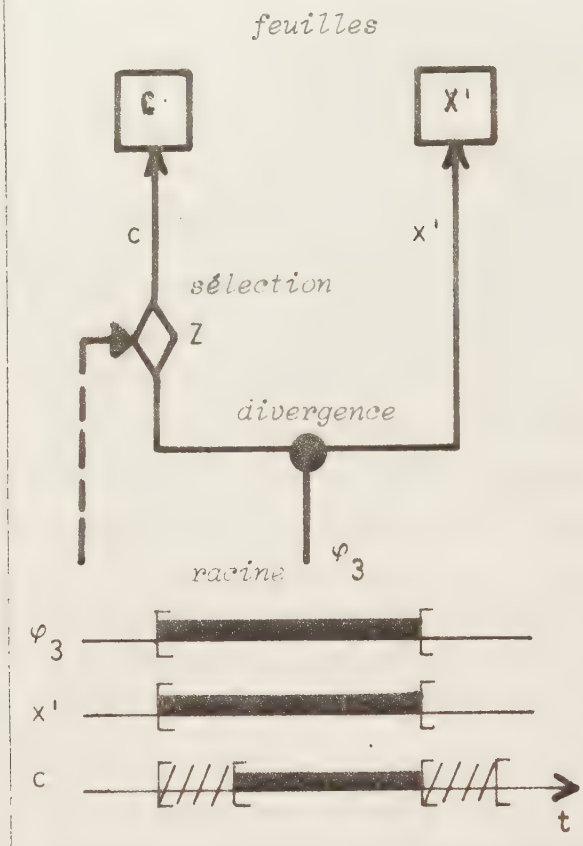
Nous avons déjà supposé, lors de la synthèse du cycle de commandes, qu'un système séquentiel est composé de deux parties :

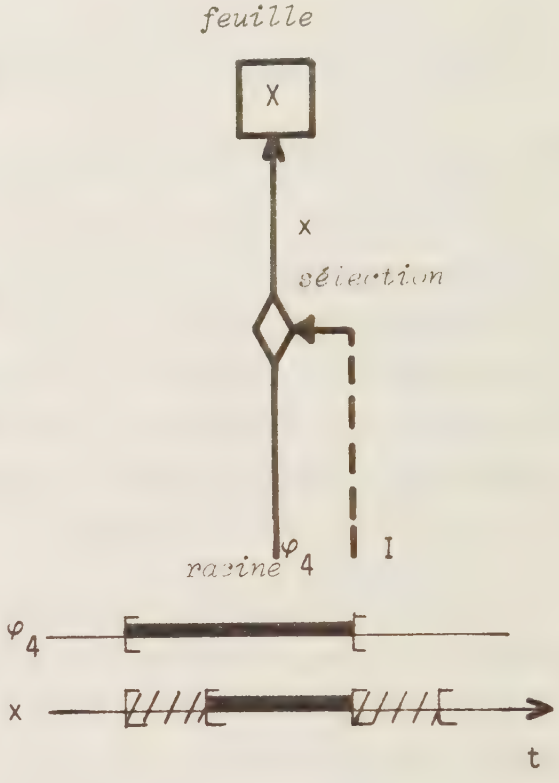
- l'horloge,
- l'ensemble des éléments de mémoire et des liaisons d'interconnexion.

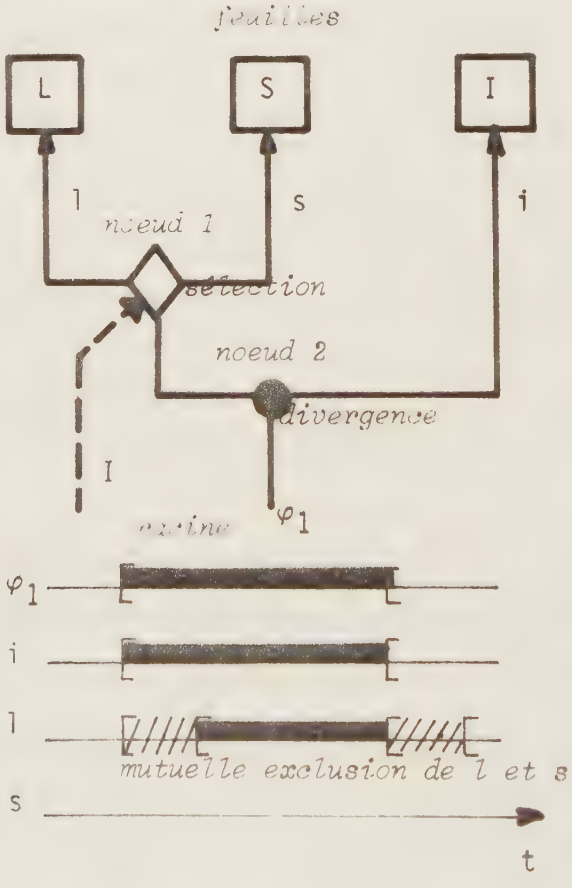
La fonction de l'horloge est de générer l'ensemble des références temporelles - ie : le système de repérage - pour la totalité du système. Cette horloge permet de repérer de façon unique tout point de transition d'un signal quelconque par rapport à un point de transition d'une "commande" temporelle.

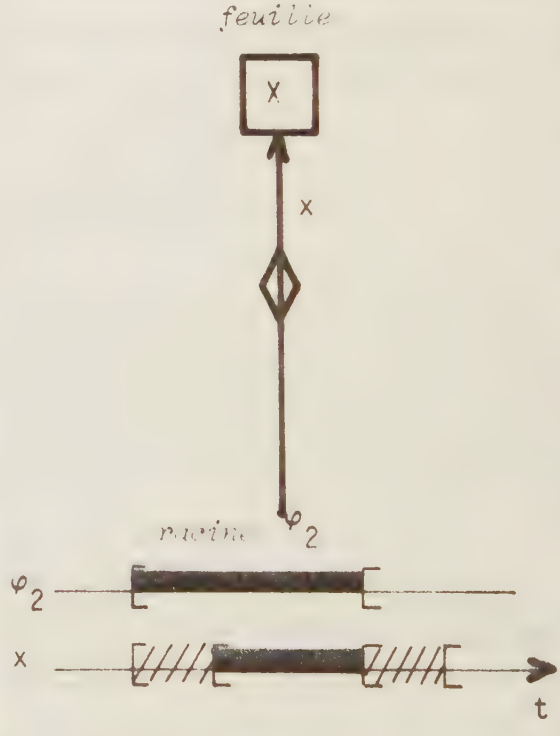
L'unicité du repérage implique qu'une commande temporelle ne peut être "bouclée" sur elle même en dehors de l'horloge, car dans ce cas le repérage d'un point de transition n'est plus unique.

La diffusion des références de temps à l'intérieur du système séquentiel a donc une structure arborescente dont les sorties de l'horloge sont les racines. Il y a en fait une arborescence à racine unique par commande.









ARBORESCENCES & ISOCHRONES

(correspondant à l'exemple précédent)

Chaque noeud de l'arborescence correspond à un éclatement de la commande en plusieurs signaux. Les éclatements se scindent en deux types :
- 1) la divergence pure : la commande est alors dupliquée en plusieurs signaux,
- 2) la sélection : à une composante de la commande initiale correspond une seule composante pour l'ensemble des signaux "images" (mutuelle exclusion temporelle). L'évaluation des contraintes doit alors être faite séparément pour chaque signal dupliqué, en fonction de la structure de l'ensemble des commandes de transfert.

La notion de transfert implique que les feuilles de l'arborescence soient obligatoirement des cellules de mémoire. En effet, seules ces cellules permettent de repérer temporellement un transfert puisqu'elles en constituent les sources et les récepteurs.

Si le point d'aboutissement d'une référence de temps est obligatoirement une cellule de mémoire, celà n'implique pas la réciproque. Ainsi, la sortie d'une première cellule de mémoire peut servir de commande temporelle à une seconde cellule ; la référence de temps "traverse" alors la première cellule pour aboutir à la seconde.

Il y a dans ce cas une "division de la fréquence du cycle" car un changement d'état de la deuxième cellule en implique deux pour la première.

Chaque arborescence délimite le domaine temporel d'une commande. Ce domaine peut être caractérisé par une isochrone dont les composantes sont les transformées sucessives d'une composante de la commande générée par l'horloge. Le nombre de composantes de l'isochrone est égal au nombre de noeuds et de feuilles de l'arborescence.

A cause des temps de retard, les coordonnées des différentes composantes ne sont pas identiques. De plus, l'incertitude sur la valeur de ces temps de retard implique que les frontières de l'isochrone ne sont pas définies exactement : elles sont bornées entre une frontière minimale et une frontière maximale.

Le cumul de ce type d'incertitude implique que les contraintes temporelles soient évaluées "localement" en tout noeud de l'arborescence. Un évaluation "globale" entraine en effet une majoration cumulative et indépendante des incertitudes sur chaque mesure temporelle. Il se peut alors que certaines contraintes ne soient pas vérifiées sur une échelle globale de mesure, alors qu'elles le sont sur une échelle locale. Nous avons étudié ce problème de repérage pour la fonction de retard (cf. annexe).

L'ordre d'évaluation des contraintes doit correspondre à un cumul progressif des incertitudes. Cet ordre correspond à une remontée de l'arborescence à partir des feuilles pour aboutir à la racine. Le calcul des contraintes est donc semblable à celui qu'on effectue en compilation pour évaluer une expression parenthèsée ; il en a toutes les propriétés et contraintes quant au nombre de calculs et variables intermédiaires [RAV].

2.3 Isochrone de contraintes

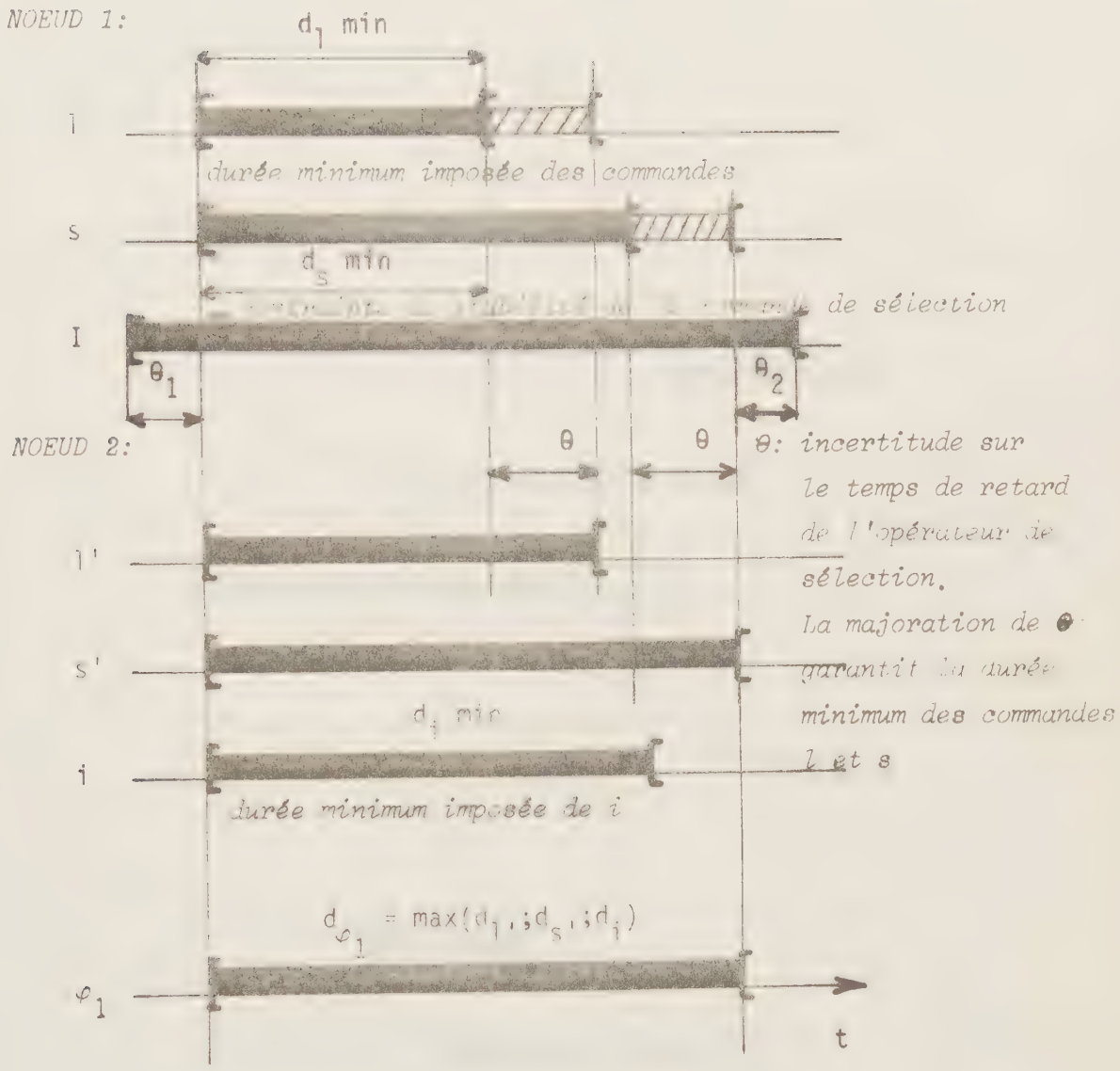
Nous avons distingué trois classes de signaux pour les cellules de mémoire :

- entrée de données,
- entrée de commande,
- sortie d'état.

L'évaluation des caractéristiques temporelles d'une cellule de mémoire a fait apparaître deux types de relations par rapport aux paramètres de la commande :

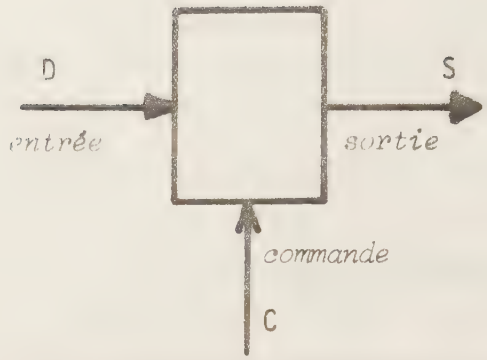
- les contraintes d'entrée (ie : de stabilité des entrées) : les entrées doivent être stables sur un intervalle de temps borné inférieurement et supérieurement ;
- la période de stabilité des sorties : cette période n'est pas bornée supérieurement, puisque la mémorisation transforme un évènement (d'établissement) en un état (propre) permanent.

Les bornes des intervalles de stabilité des entrées (ou des sorties) sont mesurées par rapport aux coordonnées temporelles (t,T) de la commande. Leur mesure permet de distinguer trois types d'intervalles, c'est à dire de contraintes :

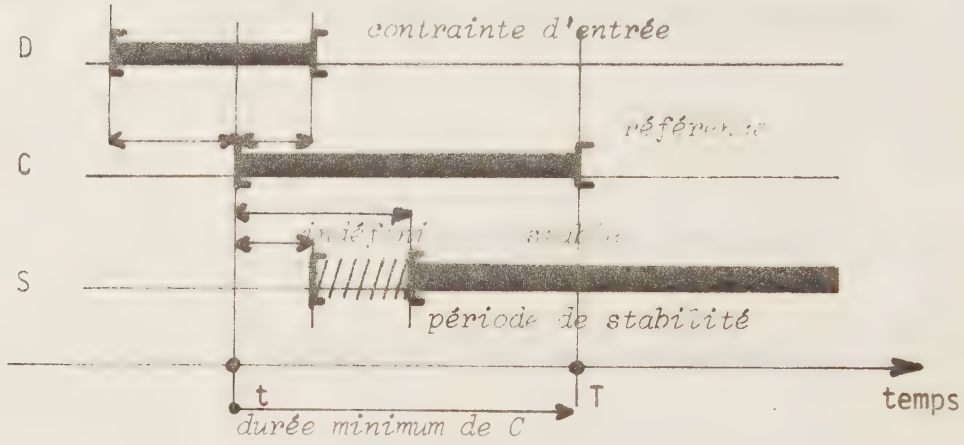


EVALUATION DE LA DUREE DE LA COMMANDE φ_1

(ordre d'évaluation: noeud 1 puis noeud 2)

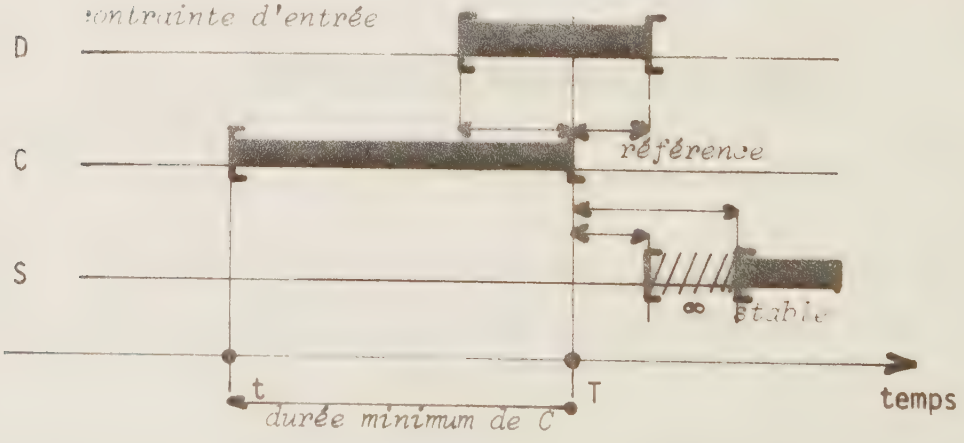


REPERAGE DES CARACTERISTIQUES TEMPORELLES DES BASCULES



1 CARACTERISTIQUES LIEES AU FRONT AVANT DE LA COMMANDE

(ex: bascule du type D / TEXAS)



2) CARACTERISTIQUES LIEES AU FRONT ARRIERE DE LA COMMANDE

(ex: bascule "maître - esclave")

- 1) contrainte liée au front avant de la commande (t) : les deux bornes sont mesurées par rapport à la date de début t ;
- 2) contrainte liée au front arrière : les deux bornes sont alors mesurées par rapport à la date de fin T ;
- 3) contrainte liée à la durée de la commande : la borne inférieure est mesurée par rapport au début (t) et la borne supérieure par rapport à la fin (T).

L'ensemble de ces mesures constitue les coordonnées de l'isochrone de contraintes associée à une des commandes d'un élément séquentiel donné ; c'est à dire en une feuille de l'arborescence de la commande.

Pour un élément séquentiel, les entrées sont généralement hiérarchisées. L'échantillonnage des données est, par exemple, distingué des commandes d'initialisation de l'élément, car l'effet de ces commandes doit être indépendant de l'état des données. Celà nécessite, lorsqu'il y a plusieurs entrées de commande, la définition d'un ensemble de contraintes d'entrée spécifique à chaque entrée de commande.

Comme l'effet logique de deux commandes indépendantes ne peut être simultané, l'analyse des caractéristiques temporelles d'une seule commande conduit alors à postuler deux types logiques de contraintes d'entrée :
- 1) mutuelle exclusion des commandes,
- 2) inclusion temporelle des données par rapport à la commande.

Cette distinction logique n'implique pas cependant une évaluation séparée de chaque type de contrainte ; car la mutuelle exclusion peut être considérée comme une inclusion temporelle, relativement au "complément temporel"(ie : le mémorisé) d'une des commandes.

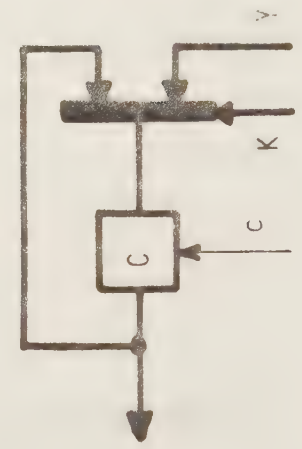
Comme les contraintes temporelles sont évaluées uniquement par rapport aux commandes de l'horloge, les contraintes d'entrée doivent être reportées sur les commandes d'horloge qui repèrent les signaux d'entrée.

Ce report des contraintes consiste alors à "remonter" tout signal d'entrée jusqu'à ses "origines temporelles". Celles - ci se scindent en deux catégories :

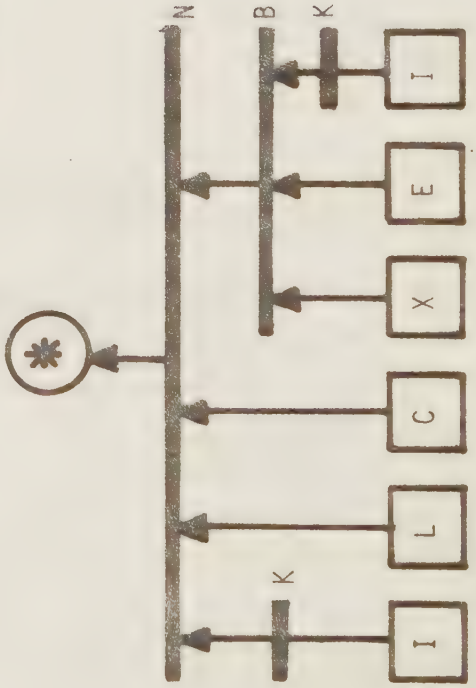
ARBORESCENCES

DES CHEMINS DE DONNEES

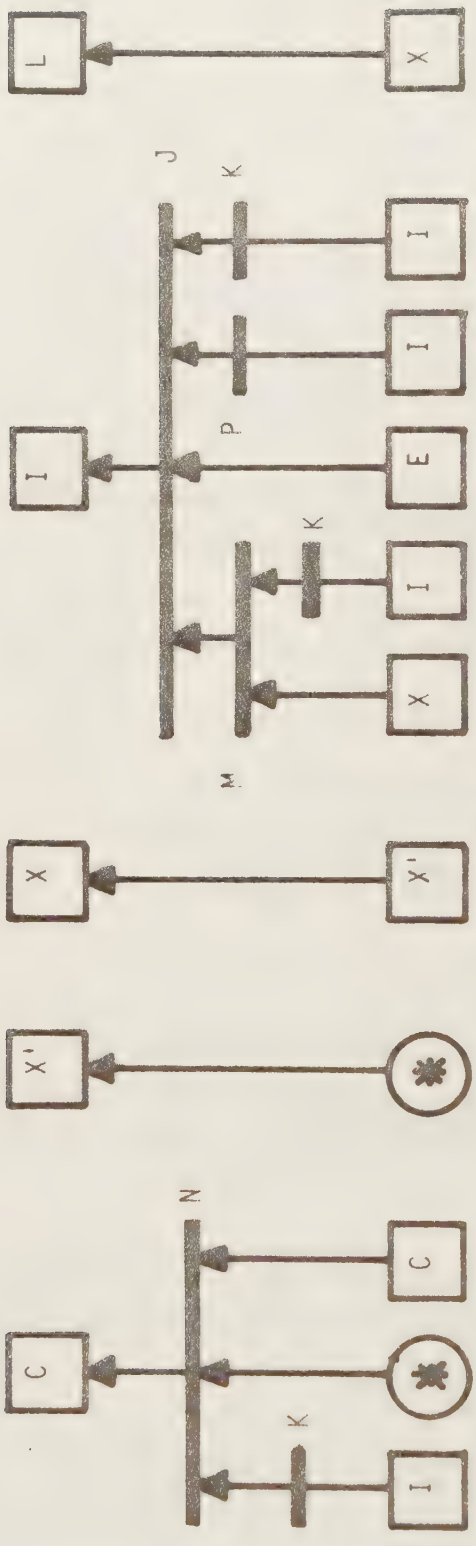
(exemple précédent)



structure du report C



sources (feuilles)



sources (feuilles)

réceptrices (racines)

- 1) les sorties de l'horloge ;
- 2) les sorties d'une cellule de mémoire : dans ce cas, les contraintes sont reportées sur les commandes temporelles de cette cellule d'origine.

La première catégorie d'origines appartient :
- 1) ou bien aux autres commandes temporelles de la même cellule de mémoire ;
- 2) ou bien à une commande de sélection systématique (multiplexage temporel) des données.

La deuxième catégorie d'origines correspond à la notion de transfert : les contraintes temporelles définies pour les récepteurs sont reportées sur les sources du transfert.

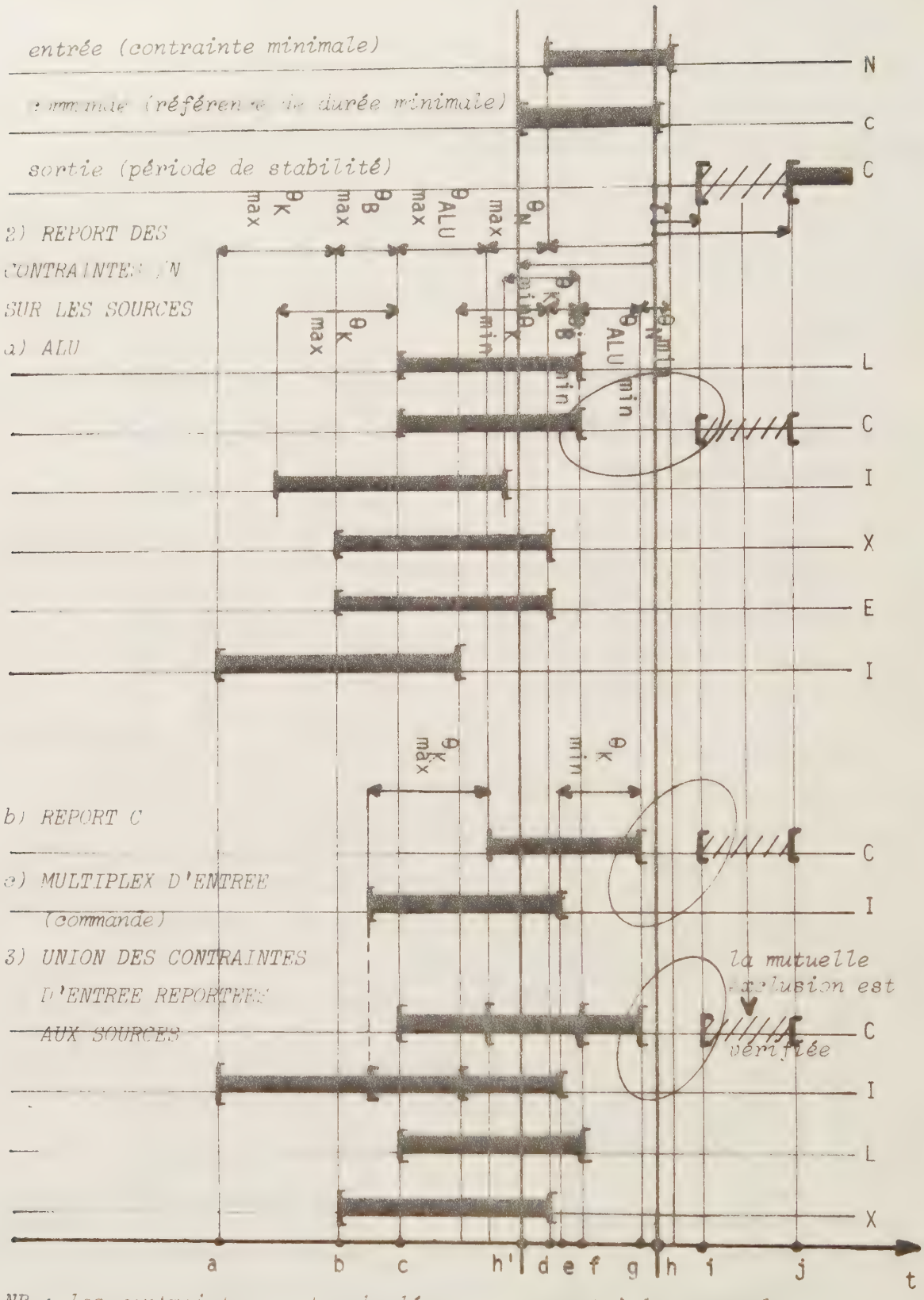
Un tel report implique donc la détermination des sources associées à une entrée de la cellule considérée comme récepteur. Ces sources sont déterminées grâce à la carte de la machine et sont regroupées d'après les clauses de sélection (ie : mutuelle exclusion temporelle) introduites lors de la définition du cycle de commandes.

Les sources sont connectées aux récepteurs par un liaison constituée d'éléments combinatoires. Cette liaison a une structure arborescente dont chaque récepteur est une racine et chaque source une feuille.

Le report des contraintes, à travers chaque élément combinatoire de la liaison, entraîne leur majoration, en fonction de l'incertitude introduite sur la valeur du temps de retard associé à l'élément combinatoire (le calcul de cette majoration est effectué en annexe).

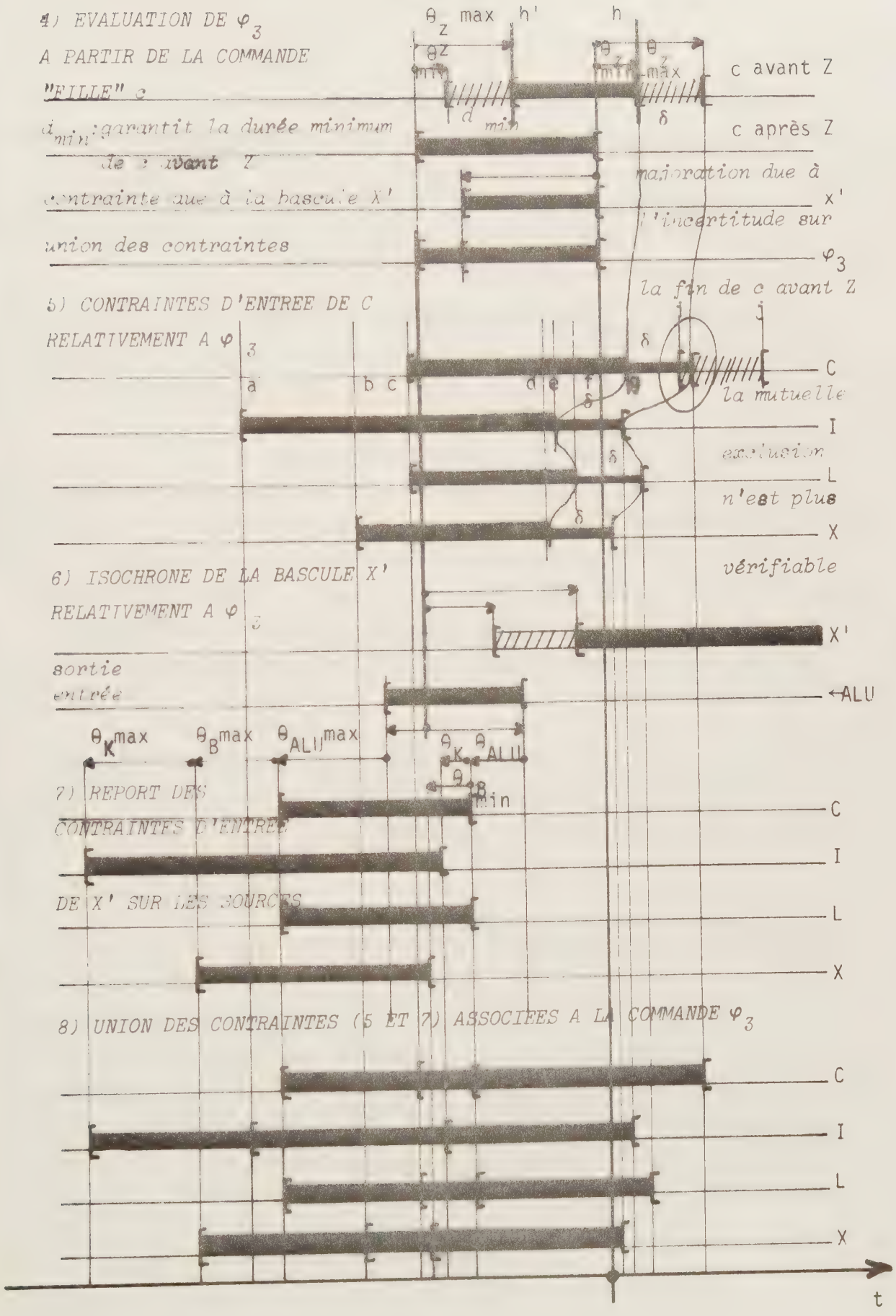
Les isochrones de contraintes associées aux feuilles d'une même arborescence doivent alors être comparées entre elles, puisqu'elles sont incluses dans le même domaine temporel.

La comparaison consiste à vérifier que les contraintes d'entrée, qui sont repérées par rapport à la commande, sont incluses dans les périodes stables de sortie ; autrement dit, celà revient à vérifier l'opacité temporelle pour l'ensemble des éléments de mémoire commandés par une même commande.



NB.: les contraintes sont calculées par rapport à la commande C avant l'opérateur de sélection Z

EVALUATION DES CONTRAINTES D'ENTREE REPORTEES AUX SOURCES DU TRANSFERT (bascule de report C de l'exemple précédent)



ISOCHRONE DE CONTRAINTES ASSOCIEE A LA COMMANDE φ3

Si l'opacité est vérifiée, les isochrones qui ont été comparées sont alors remplacées par leur union, afin d'obtenir la majoration des contraintes.

La comparaison est d'abord effectuée pour chaque feuille, c'est à dire pour chaque cellule de mémoire contrôlée par la commande. Cette comparaison permet de vérifier l'opacité de la cellule "bouclée" sur elle même.

L'isochrone des contraintes associée à chaque feuille est alors reportée sur les noeuds dont la feuille est issue. Ce report entraine une majoration due à l'incertitude introduite par les temps de retard des éléments traversés par le signal de commande.

Lorsque deux isochrones de contraintes sont reportées sur un même noeud, elles sont comparées entre elles uniquement dans le cas où le noeud correspond à une divergence ; car dans le cas d'un noeud de sélection, ces deux isochrones sont mutuellement exclusives dans le temps . L'union des isochrones est cependant effectuée dans les deux cas, afin d'obtenir une majoration des contraintes. L'isochrone résultante est alors reportée vers la racine de l'arborescence.

Finalement, l'isochrone globale des contraintes associées à une commande est ainsi obtenue par union sucessive des isochrones de contraintes associées à chaque feuille.

2.4. Concaténation des domaines temporels

Un cycle de commandes est constitué d'un ensemble de séquences de commandes mutuellement exclusives dans le temps. A chaque commande nous venons d'associer un isochrone (globale) de contraintes qui délimite - relativement à chaque signal du système - les intervalles de temps pendant lesquels les autres commandes du cycle doivent être neutres, c'est à dire : ne doivent pas interférer avec cette commande.

Cette isochrone de contraintes spécifie complètement le domaine temporel d'une commande. Les bornes inférieures et supérieures de son paramètre de durée délimitent l'ensemble des valeurs possibles de la durée de la commande.

Il reste à déterminer les positions temporelles extrêmes (ie : les dates) de cette commande par rapport à celle des autres commandes de la séquence du cycle à laquelle cette commande appartient.

Comme les commandes d'une même séquence sont générées l'une après l'autre, il suffit d'évaluer la position de la commande par rapport à ses prédécesseurs ou bien par rapport à ses successeurs (celà dépend de l'ordre d'évaluation - croissant ou décroissant - de la séquence de commandes).

La position de la dernière commande de la séquence doit être évaluée par rapport à la première de chacune des séquences successeurs de cette séquence.

Les bornes inférieures et supérieures de l'intervalle de temps (ie : intervalle de position), qui sépare le début de deux commandes successives, sont obtenues en effectuant la concaténation des isochrones de contraintes associées à ces commandes.

Chaque isochrone de contraintes peut être décomposée en une isochrone d'entrée, qui regroupe toutes les contraintes d'entrée, et en une isochrone de sortie qui délimite les périodes de stabilité des sorties.

Dans ce cas, l'isochrone d'entrée de la commande - successeur doit être incluse dans l'isochrone de sortie de la commande - prédécesseur.

La borne inférieure de la durée de l'intervalle de position correspond au cas où les frontières inférieures
- 1) de l'isochrone d'entrée du successeur
- 2) et de l'isochrone de sortie du prédécesseur, sont jointives.

Inversement, la borne supérieure correspond au cas où les frontières supérieures des mêmes isochrones sont jointives.

Comme les frontières des isochrones sont mesurées - ou bien par rapport à la date de début - ou bien par rapport à la date de fin des commandes, les bornes de l'intervalle de position dépendent de la durée des commandes.

Un cas particulier remarquable est celui où toutes les commandes ont la même durée et où les intervalles de position sont des multiples d'une période de base. L'horloge peut alors être remplacée par une horloge biphasée unique et un automate séquentiel.

2.5. Superposition de séquences de commande

Il arrive parfois (par exemple, pour la commande des mémoires à tores) que le cycle de commandes ne puisse pas être linéarisé (ie : réduit) en une séquence unique de commandes temporelles. Plusieurs commandes "se chevauchent" alors temporellement ; de façon plus précise, l'intersection des repères associés à ces commandes n'est pas vide sur la période du cycle de commandes.

L'évaluation des contraintes - pour chaque séquence de commandes - est alors identique à celle qui vient d'être décrite. A chacune des séquences est alors associée l'isochrone de contraintes obtenues par concaténation des isochrones associées aux commandes.

L'évaluation globale des contraintes est alors obtenue en effectuant l'union temporelle des isochrones associées aux séquences.

La compatibilité des différentes contraintes est vérifiée si la concaténation des isochrones d'entrée de chaque séquence est incluse dans celle des isochrones de sortie de chacune des autres séquences.

III - DOMAINES TEMPORELS

A l'aide des signaux de commande qu'elle génère, l'horloge délimite le domaine temporel du système séquentiel ; en particulier, elle détermine la durée des commandes et les écarts de temps qui séparent les différentes commandes. Lors de l'évaluation d'un diagramme des temps, un"degré de liberté" a été conservé par le fait que les relations qui lient ce domaine temporel à ceux qui sont externes au système séquentiel, ont été négligées. Nous avons justifié cette simplification en admettant que les échanges (ie :"Entrées/ sorties") avec l'extérieur pouvaient être considérés comme des transferts élémentaires internes. L'étude des relations qui lient entre eux plusieurs domaines temporels va permettre de vérifier cette hypothèse.

3.1. Domaines synchrones et asynchrones

Deux types de relations temporelles sont en général distinguées :

- les relations synchrones,
- les relations asynchrones.

Nous avons vu, dans le chapitre I.4, que cette classification correspond à celle de l'origine du repérage temporel : - interne ou - externe au système. Il importe, pour la suite, de définir plus précisément ces notions de synchronisme et d'asynchronisme pour des domaines temporels.

Domaine synchrone : un domaine temporel est synchrone dans la mesure où il est fermé. L'ensemble des signaux associés à un domaine synchrone est tel que la cause temporelle de tout point de transition d'un de ses signaux est un autre point de transition d'un signal de l'ensemble.

Domaines asynchrones : la notion d'asynchronisme correspond, ici, à la partition d'un seul domaine temporel synchrone en plusieurs domaines disjoints. Chacun de ces domaines est ouvert, ie : sa définition temporelle est partielle. La définition complète des relations temporelles ne peut alors être faite qu'à l'aide de l'ensemble des domaines temporels ouverts.

Une telle partition de domaines"asynchrones"conduit à une notion du temps relative et locale. A proprement parler, seule l'union de domaines synchrones disjoints est en fait asynchrone (dans le sens commun du terme).

Dans tous les cas, la référence de temps associée à un domaine temporel est caractérisée par l'ensemble des signaux de commande temporels et elle peut être définie à l'aide des repères partiels dont les composantes définies délimitent les périodes d'activation des commandes.

Pour un domaine temporel synchrone, la référence de temps est une suite unique de dates qui sont liées à leur successeur et prédécesseur par des relations de cause à effet. Ces relations sont fonctionnelles : à un instant donné un seul évènement définit le temps.

Par contre, pour un ensemble (synchrone) de domaines temporels asynchrones, la référence de temps commune est l'intersection des références partielles associées à chaque domaine asynchrone. Chaque date de la référence commune n'a qu'une seule cause temporelle définie par une date d'une des références partielles. Pour l'ensemble des domaines asynchrones, la définition du temps est en effet fonctionnelle, puisque cet ensemble délimite un domaine temporel synchrone. Cependant, entre deux dates de la référence commune, la définition du temps n'est pas unique, elle est relative à chaque domaine asynchrone.

Il résulte de nos définitions que les relations qui lient deux domaines synchrones ne peuvent porter que sur la succession des évènements. Elles ne peuvent porter sur les dates, puisque celles - ci sont définies à partir de domaines temporels fermés. Nous disons dans ce cas, que les domaines synchrones sont couplés logiquement, alors qu'ils sont découplés temporellement.

Par contre, deux domaines asynchrones sont couplés temporellement, puisqu'ils définissent un seul et même domaine temporel synchrone.

3.2. Couplage

La seule façon de coupler deux domaines temporels est de "retarder le temps" d'un des domaines, afin de le synchroniser sur le temps relatif à l'autre domaine.

La référence de temps partielle de chaque domaine asynchrone peut être considérée comme une suite de séquences d'évènements, où le premier évènement de chaque séquence caractérise une date de la référence de temps commune (point de synchronisation).

D'autre part, nous avons exposé, dans la note du chapitre I.2, comment les lois d'union et d'intersection peuvent s'exprimer en termes d'évènements.

Dans ces termes, l'opération de couplage s'exprime comme l'union des derniers évènements, pour chaque séquence caractéristique d'un domaine asynchrone*. La plus grande des dates de fin de ces évènements détermine la date du début du premier évènement de la séquence suivante. Comme cette date de début doit être unique, la durée des derniers évènements doit être telle que leur union doit délimiter un évènement résultant unique. Par hypothèse, aucune relation temporelle ne lie les dates de fin entre elles ; il en résulte que l'opération de couplage nécessite la mémorisation des évènements. Cependant la mémorisation transforme un évènement en un état dont la durée n'est pas bornée ; la durée de mémorisation doit alors être limitée par la présence d'un évènement d'une autre référence de temps partielle.

La relation complète de couplage se transforme donc en une double relation d'"asynchronisme" qui peut être étudiée à l'aide du couple de signaux de requête (READY) et d'accusé de réception (ACKNOWLEDGE) de la théorie des mécanismes d'échanges asynchrones [PAT] (cette étude sur la durée des signaux n'est pas faite, à notre connaissance, dans cette théorie).

Le principe de base est que l'émetteur génère un signal de requête vers le récepteur qui répond à ce signal en générant un accusé de réception.

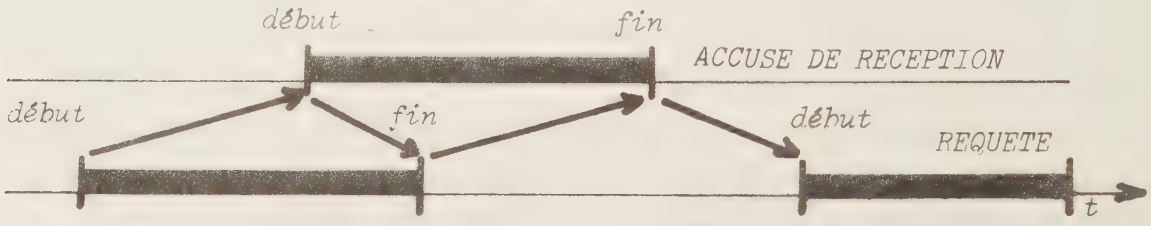
Avec notre terminologie, la requête et l'accusé de réception sont des évènements dont la durée dépend du "temps de réponse" du destinataire. Ainsi, la requête doit être maintenue jusqu'à l'apparition de l'accusé de réception. De même, cet accusé de réception doit être maintenu jusqu'à ce que l'émetteur de la requête en prenne connaissance.

Il résulte de ces contraintes qu'il n'y a qu'un mécanisme temporel possible pour cette procédure de couplage :

1) le début de l'accusé de réception a pour origine le début de la requête

N.B.: Si l'on considère les repères complémentaires, dont les composantes définies délimitent une absence d'évènement, la loi d'union doit alors être remplacée par celle d'intersection; la mémorisation (ie: complémentation temporelle) permet de passer de l'une à l'autre loi (cf. note du chapitre I.2).

(avec un "temps de réponse" indéterminé, propre au récepteur de la requête) ;
- 2) la fin de la requête a pour origine le début de l'accusé de réception (avec un temps de réponse propre à l'émetteur) ;
- 3) la fin de l'accusé de réception a pour origine la fin de la requête (avec un temps de réponse propre au récepteur) ;
- 4) le début de la prochaine requête a pour origine la fin de l'accusé de réception précédent (avec un retard propre à l'émetteur).



Cet enchaînement logique des évènements, que sont le début et la fin des deux signaux, doit être soigneusement distingué de la suite des relations temporelles de cause à effet. Les confondre revient à introduire des relations temporelles supplémentaires.

En effet, chaque évènement, qu'est le début ou la fin d'un des deux signaux, a une cause dont la durée n'est pas nulle. Il se peut alors que la cause de la fin de la requête - qui serait alors le début de l'accusé de réception - apparaisse avant la fin de la cause du début de cette requête. Cette situation est contradictoire ; pour être évitée, la cause de la fin de la requête doit résulter d'une combinaison des deux références de temps. Ainsi, elle doit succéder à la fin de la cause de son début (référence "interne") et au début de l'accusé de réception (référence de temps "externe").

Le formalisme des graphes de Pétri[DEN] masque ce problème. Ainsi, l'insertion ou la suppression d'un point dans un cercle, qui concrétisent la présence ou l'absence d'un évènement, sont supposées "instantanées". Celà implique que la cause et l'effet se succèdent et sont mutuellement exclusifs dans le temps ; le problème soulevé ne peut donc être observé.

La relation de couplage qui vient d'être décrite correspond au mécanisme

complet de synchronisation temporelle. L'opération de couplage peut en fait s'étaler sur toute la durée des séquences d'évènements propres à chaque domaine asynchrone, c'est à dire entre "deux points de synchronisation".

Nous avons vu que le couplage est effectué à l'aide des opérations de mémorisation et d'union (ou d'intersection). Le mécanisme associé doit comprendre au minimum des fonctions extrêmales, car les éléments de mémoire peuvent être omis lorsque les temps de réponse de l'émetteur et du récepteur sont majorés. Il suffit alors que la durée des évènements soit supérieure à ces temps de réponse. Une telle possibilité implique une relation temporelle sur la mesure du temps pour chaque domaine asynchrone.

Les mécanismes de couplage associés à un ensemble de domaines temporels asynchrones constituent en fait l'horloge du domaine synchrone résultant. Vouloir doter - comme c'est le but de la théorie des sytèmes asynchrones - chaque élément d'un système séquentiel d'un mécanisme de couplage revient à répartir l'horloge du système. Une telle répartition ne se justifie - à notre avis - que pour les structures physiquement grandes, par l'importance des temps de propagation des signaux. Sur une échelle physique réduite, une telle structure ne peut apporter un gain ni sur le nombre de composants ni sur la "vitesse", car le calcul du diagramme des temps permet d'évaluer l'horloge "centralisée" optimale.

3.3. Découplage temporel

Par définition, il ne peut y avoir aucune relation temporelle entre deux domaines synchrones disjoints, puisque ces domaines sont fermés. Il en résulte que la seule opération, qui permette de faire coïncider l'émission et la réception d'un signal entre deux domaines synchrones, est la mémorisation. Ainsi, l'émetteur doit mémoriser l'information, jusqu'à ce que le récepteur l'acquiert.

A l'opposé, l'acquisition de l'information n'a de sens que si celle - ci est stable (ie : dans un état défini) lors de l'échantillonnage. Malheureusement, cette contrainte ne peut être vérifiée a priori, puisqu'il n'y a aucune relation temporelle entre les domaines. Il apparait alors que la principale fonction du mécanisme de découplage temporel est de déterminer - relativement à la référence de temps du récepteur - les périodes pendant lesquelles le signal, transmis par l'émetteur, est stable.

Le mécanisme de découplage a ainsi deux fonctions :
- 1) transmission de l'information proprement dite ("données"),
- 2) reconstitution d'une référence de temps associée à l'information transmise.

La deuxième fonction ne peut être réalisée que par un mécanisme logique, puisque - rappelons le - aucune relation temporelle ne lie les domaines émetteur et récepteur. La référence de temps de la donnée transmise est alors concrétisée par un système de sémaphores entre les domaines émetteur et récepteur.

Le principe de fonctionnement du système de sémaphores est fondé
- 1) d'une part, sur l'échantillonnage "permanent" et
- 2) d'autre part, sur la propriété de complémentation temporelle des repères, de la commande d'échantillonnage et de la sortie d'état, d'un élément de mémoire.

Sémaphore uni-directionnel

Le sémaphore nécessite deux éléments de mémoire ; le "maître" et "l'esclave" :
- 1) le maître est commandé par l'émetteur,
- 2) l'esclave est commandé par le récepteur et est connecté à la sortie d'état du maître.

L'élément de mémoire maître a pour fonction de mémoriser l'état du sémaphore jusqu'à ce que le récepteur en prenne connaissance.

L'élément de mémoire esclave a pour fonction de délivrer un signal stable vers le récepteur. L'état de l'esclave est pris en compte par le récepteur, en dehors des périodes d'échantillonnage définies sur sa propre référence de temps. Cet élément de mémoire doit posséder une propriété physique fondamentale : il doit filtrer les oscillations (cf. chapitre II.2) qui peuvent provenir de la coïncidence de l'échantillonnage avec un changement d'état du maître.

Le principe de fonctionnement du sémaphore consiste en ce que le récepteur commande en permanence (ie : périodiquement) l'échantillonnage du maître par l'esclave, jusqu'à ce qu'un changement de son état soit détecté.

Sémaphore double

Un seul sémaphore ne suffit pas, car il se pourrait que l'émetteur change l'état du maître à une fréquence supérieure à celle d'échantillonnage de l'esclave. Deux sémaphores sont donc nécessaires. Le principe du couplage logique est celui qui a été étudié pour le couplage temporel ; un sémaphore symbolise la requête

et l'autre l'accusé de réception.

Le principe de fonctionnement des sémaphores est alors le suivant ,

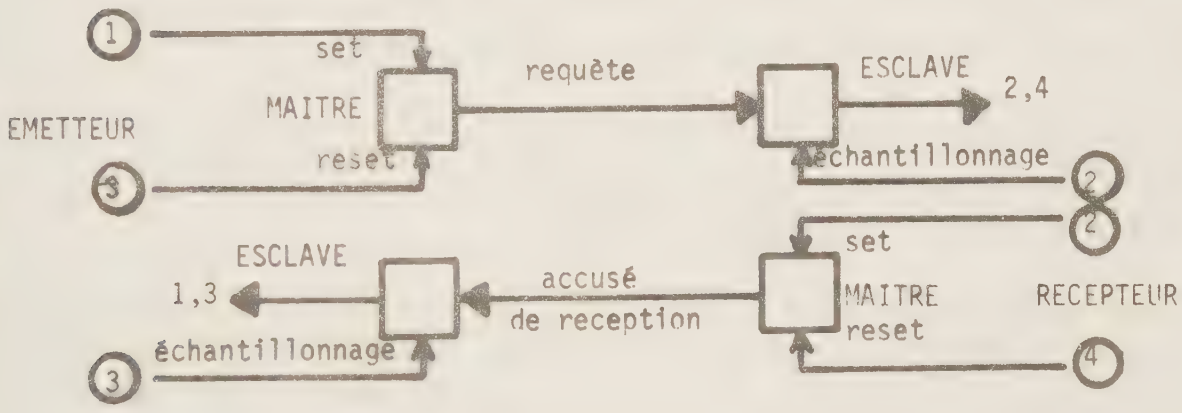
- 1) L'émetteur établit le sémaphore maître de requête ;
- 2) Le récepteur échantillonne en permanence l'esclave correspondant ;

Lorsque le récepteur détecte une requête, il établit le sémaphore maître d'accusé de réception ;

- 3) Depuis l'établissement de la requête, l'émetteur échantillonne en permanence l'esclave correspondant à l'accusé de réception.

Lorsqu'un accusé de réception est détecté, l'émetteur remet le sémaphore maître de requête dans son état de référence ;

- 4) Depuis l'établissement de l'accusé de réception, le récepteur échantillonne en permanence l'esclave du sémaphore de requête. Lorsque la fin de la requête est détectée, le sémaphore maître d'accusé de réception est remis dans son état de référence.



Echanges

Le système de sémaphore double permet de reconstituer, sur l'échelle de temps du récepteur, la référence de temps associée à la donnée à échanger. La transmission de cette donnée nécessite alors uniquement un seul élément de mémoire tampon, puisque le sémaphore double permet de garantir la mutuelle exclusion d'accès à ce tampon, entre le récepteur et l'émetteur.

Les échanges se scindent en deux catégories, selon que la requête provient de l'émetteur ou bien du récepteur (l'expéditeur de la requête est en fait "le maître de l'échange"). Les procédures d'échange diffèrent légèrement dans chaque cas :

1) Emetteur maître de l'échange

1 - établissement du tampon de données ;
2 - requête issue de l'émetteur ;
3 - Prise en compte de la requête et acquisition de la donnée par le récepteur ;
4 - accusé de réception issu du récepteur.

2) Récepteur maître de l'échange

1 - requête issue du récepteur ;
2 - prise en compte de la requête et établissement du tampon de donnée par l'émetteur ;
3 - accusé de réception issu de l'émetteur ;
4 - acquisition de la donnée par le récepteur.

Tout échange d'information, ne serait-ce que d'un seul bit d'information requiert donc deux sémaphores, plus un élément de mémoire tampon pour la donnée à transmettre, soit au minimum cinq éléments de mémoire !

La complexité de cette procédure de couplage logique est telle qu'elle n'est jamais employée pour des échanges élémentaires, elle n'est utilisée - à la rigueur - que pour des transferts de paquets d'information.

En général, des relations temporelles "locales" supplémentaires sont introduites afin de simplifier le mécanisme d'échange. Pour les machines séquentielles, la classification des types de couplage - logique ou temporel - est approximativement la suivante :
- 1) couplage temporel entre deux domaines asynchrones : limité à des opérateurs d'un même organe interne à la machine,
- 2) couplage entre un domaine (principal) synchrone et des domaines (secondaires) asynchrones : utilisé pour les échanges entre l'unité centrale et les unités périphériques (principe du "START-STOP");
- 3) couplage entre plusieurs domaines synchrones : c'est l'apanage des échanges entre machines séquentielles ; le système de sémaphore constitue alors le système d'interruptions (au sens large : cablé et programmé).

Quelquesoit la structure envisagée, le principe reste toujours le même : le découplage temporel nécessite dans tous les cas des éléments de mémoire physiquement indépendants qui doivent filtrer les oscillations.

- STRUCTURE D'UNE HORLOGE

Nous distinguons deux types d'horloges : - les horloges "monophasées" et - horloges "multiphasées".

1. Horloges monophasées

Le "coeur" de ce type d'horloge est un oscillateur (cf. Note 1) qui génère seul signal de commande, "binaire" et périodique, dont un des deux états délite les périodes d'activation de cette commande.

Toutes les autres commandes correspondent alors aux sorties d'un automate états, dont la commande d'activation est le signal d'horloge. Toutes les commndes ont alors une durée et une période multiples de la période du signal horloge.

Autrement dit, toutes les caractéristiques temporelles des commandes sont rivées de celles du signal unique d'horloge. Puisque ces caractéristiques sont mmunes à toutes les commandes elles sont obligatoirement majorées - ou du moins antifiées - afin d'être compatibles avec les contraintes imposées par l'ensemble s éléments logiques. Il en résulte que les performances optimales ne peuvent re atteintes. De plus, une modification des caractéristiques temporelles d'un ément implique une re-définition de l'automate d'état (par exemple, remplacement une mémoire à tores par une mémoire à circuits intégrés).

Notons, par contre, que les horloges monophasées sont les plus simples et les us faciles à réaliser physiquement.

2. Horloges multiphasées

Une horloge de ce type génère un ensemble de commandes dont chacune a ses ropres caractéristiques temporelles. Chaque commande correspond à une "phase". a multiplicité des phases résulte uniquement de contraintes physiques. Plus difficiles à réaliser, ces horloges permettent cepandant d'atteindre les caractéristiques optimales. Elles permettent en particulier d'adapter les caractéristiques emporelles descommandes aux temps de propagation des signaux.

Nous nous intéressons uniquement à ce second type d'horloge.

Le fondement des horloges multiphasées est l'élément monostable. Ce type 'élémcnt transforme un évènement ponctuel, déterminé par un changement d'état du ignal d'entrée (ie : transition), en un état de durée finie et invariante. Cette

durée, qui est une caractéristique de l'élément monostable, sert d'étalon de mesure du temps. Elle détermine - ou bien la durée d'une commande - ou bien l'intervalle de temps qui sépare deux commandes.

Cet état temporaire de l'élément monostable permet de concrétiser un "évènement". La structure de l'horloge peut alors être décrite comme une combinaison de suites d'évènements, à l'aide des lois d'union, d'intersection, de mémorisation et de la relation de transfert.

Pour des éléments monostables, la relation de transfert associe logiquement une "date" et un état. Il faut alors distinguer les dates de début et de fin d'un évènement, ainsi que la présence de l'absence d'un évènement. Pour celà, nous avons adopté les conventions suivantes :

Etats : Une lettre majuscule dénote un repère dont les composantes caractéristiques délimitent les évènements. Une lettre minuscule dénote le repère complémentaire, c'est à dire dont les composantes caractéristiques délimitent les absences d'évènements.

Dates : Le graphisme $E \leftarrow C\uparrow$ indique que le début d'un évènement repéré par C est la cause d'un évènement repéré par E.
Au contraire, le graphisme $E \leftarrow C\downarrow$ indique que la fin d'un évènement de C (ie : le début de l'absence d'évènement) est la cause d'un évènement de E. Les expressions $E \leftarrow C\downarrow$ et $E \leftarrow c\uparrow$ sont équivalentes.

Mémorisation : Le graphisme $S \leftarrow E/m$ dénote la mémorisation. Le début de l'état S a pour cause le début de E (établissement) et la fin de l'état S a pour cause la fin de m (maintien).

4.3. Exemple

Nous reprenons l'exemple étudié lors de la définition d'un cycle de commandes et du diagramme des temps associé.

Dans cet exemple, nous n'avons pas encore défini les caractéristiques du mécanisme d'échanges. Tel qu'il est spécifié, le système décrit peut être - ou bien synchrone - ou bien asynchrone. Nous allons envisager les deux cas.

SYSTEME SYNCHRONE

D'après nos définitions, il n'y a alors aucune relation temporelle entre l'horloge du système et l'extérieur. Dans ce cas, plusieurs éléments de mémoire tampons doivent être ajoutés en entrée et en sortie :

Entrée : - tampons des données D_e,
- Sémaphores esclaves - de requête d'entrée (R_e),
- d'accusé de réception (A_s)
(pour les sorties),

Sortie : - tampons des données (D_s),
- Sémaphores maîtres - de requête de sortie (R_s),
- d'accusé de réception (entrée) (A_e).

En entrée, les éléments de mémoire doivent être échantillonnés avant chaque transfert dont une des origines est une entrée ; c'est à dire avant la commande x' qui contrôle la doublure du tampon de sortie (X) de la boîte à opérations (phase P3). Les caractéristiques temporelles de la commande d'échantillonnage sont dans ce cas définies par la phase P2 ; la commande logique résulte de l'intersection de cette phase P2 avec une sortie d'état du circuit de commande ROM.

En sortie, les tampons de découplage sont commandés par la phase P1 (commande s), comme s'il s'agissait d'un transfert interne.

La solution proposée pour le mécanisme d'échanges implique que le système est maître des échanges de sortie, alors qu'il est "esclave" des échanges d'entrée.

Une telle structure, purement synchrone, implique que les échanges sont entièrement (micro) programmés. Les algorithmes précédemment décrits doivent alors être "déroulés par programmes".

SYSTEME ASYNCHRONE

Afin d'éviter le déroulement programmé de la procédure de couplage logique, le mécanisme de couplage peut être intégré dans l'horloge. Dans ce cas, la référence de temps du système est liée à l'extérieur - lors des échanges - : de logique, le couplage devient temporel.

Entrée : Avant de déclencher un transfert dont une des origines est une entrée E, l'horloge doit générer une requête et attendre l'accusé de réception qui indique que la donnée est stable et peut être échantillonnée.

Comme les transferts de ce type sont commandés par la phase P3 (établisse-

ment de la doublure du tampon X), la requête et l'attente doivent précéder cette phase. De plus, elles doivent succéder à la phase P1 qui commande un changement d'état du système (établissement du registre d'adresse I du circuit de commande ROM). D'autre part, l'horloge ne peut retirer la requête que lorsque le transfert est effectué, c'est à dire après la phase P3 et avant la phase P1 suivante. Il en résulte que la phase P3 est intercalée entre le début de l'accusé de réception et la fin de la requête.

Sortie : Les transferts, qui ont une sortie S pour destination, ont pour unique origine le registre tampon X, dont le contenu est significatif après la phase P4. La requête et l'attente de l'accusé de réception doivent donc succéder à la phase P4. De plus, elles doivent précéder ou coïncider avec la phase P1 qui commande - sur son front arrière - les changements d'état du système (le registre I est du type maître - esclave). D'autre part, la donnée à transmettre n'a pas besoin d'être "tamponnée", puisqu'elle est mémorisée dans le registre tampon X, à partir de la phase P4 et jusqu'à la phase P2 suivante.

Structure de l'horloge

Pour les raisons exposées dans la note sur les oscillateurs, une structure d'horloge "en étoile" est choisie.

Pour celà, le cycle est divisé en deux périodes principales :
- 1) la première (H1) regroupe les phases P1 et P2,
- 2) la seconde (H2) regroupe les phases P3 et P4.

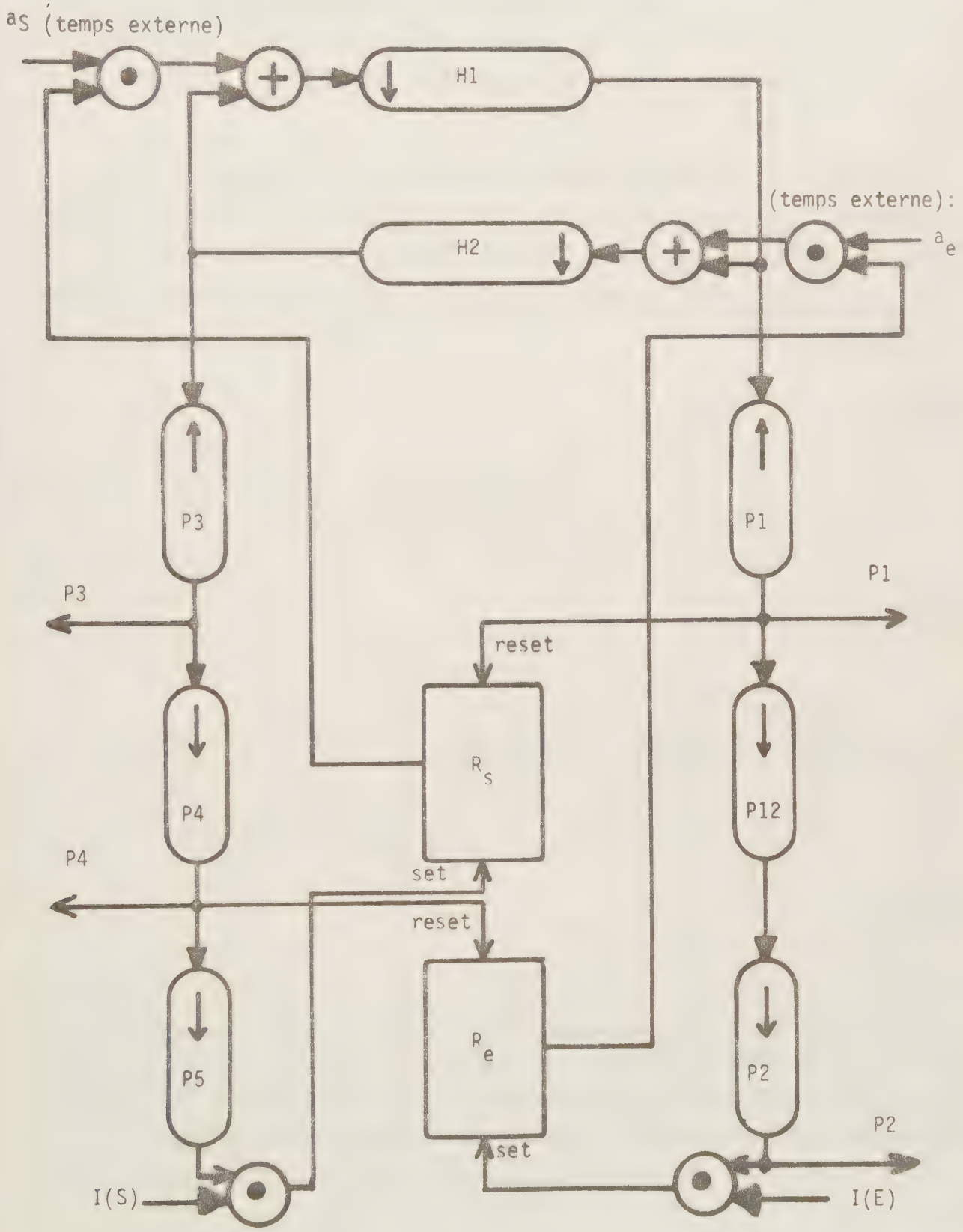
La synchronisation des entrées/sorties se fait au début de chaque période :
- entrées : déclenchement (et retardement) de la période H2.
- sorties : déclenchement (et retardement) de la période H1.

L'échelle de temps est déterminée par deux éléments monostables qui définissent la durée "normale" (ie : lorsqu'il n'y a pas d'entrée/sortie) des périodes H1 et H2. Des éléments monostables supplémentaires déterminent la durée et la position des phases P1, P2, P3, P4 relativement au début des périodes H1 et H2.

L'horloge est maître des échanges, quelquesoit leur sens : en entrée et en sortie. Deux sémaphores maîtres de requête sont alors commandés par l'horloge :

Entrée : Le sémaphore de requête (R_e) est établi ("Set") par la phase P2 et remis

HORLOGE MULTIPHASEE "ASYNCHRONE"



dans son état de référence par la phase P4.

Sortie : Le sémaphore de requête (R_S) est établi par une phase supplémentaire P5, qui succède à la phase P4, car le sémaphore est une bascule transparente dont la sortie d'état ne peut varier qu'à la fin de la phase P4. Ce sémaphore est remis dans son état de référence par la phase P1.

L'établissement des deux sémaphores est contrôlé par une sortie d'état (I(E) ou I (S)) du circuit de commande ROM, car il n'a lieu que pour une entrée/sortie. Par contre, la mise dans l'état de référence peut être systèmatique.

Avec nos conventions, les équations caractéristiques de l'horloge sont alors les suivantes.

Périodes :

$$H1 \leftarrow (H2 + R_S.a_S) \downarrow$$
$$H2 \leftarrow (H1 + R_e.a_e) \downarrow$$

(a_S et a_e dénotent les repères complémentaires des accusés de réception)

Phases :

Période H1 :

$P1 \leftarrow H1 \uparrow$	durée de la phase P1
$P12 \leftarrow P1 \downarrow$	intervalle entre les phases P1,P2
$P2 \leftarrow P12 \downarrow$	durée de la phase P2

Période H2 :

$P3 \leftarrow H2 \uparrow$	Les phases P3, P4, P5
$P4 \leftarrow P3 \downarrow$	sont consécutives
$P5 \leftarrow P4 \downarrow$	(P5 : : = commande du sémaphore R_S)

Sémaphores :

Sortie : $R_S \leftarrow P5.I(S)/p_1$

Entrée : $Re \leftarrow P2.I(E)/p_4$

Pour conclure cet exemple, nous remarquons que la structure "asynchrone" proposée a un inconvénient majeur : l'horloge est bloquée tant qu'un accusé de réception n'est pas renvoyé après une requête : le destinataire des échanges définit ainsi partiellement le "temps propre" à cette horloge.

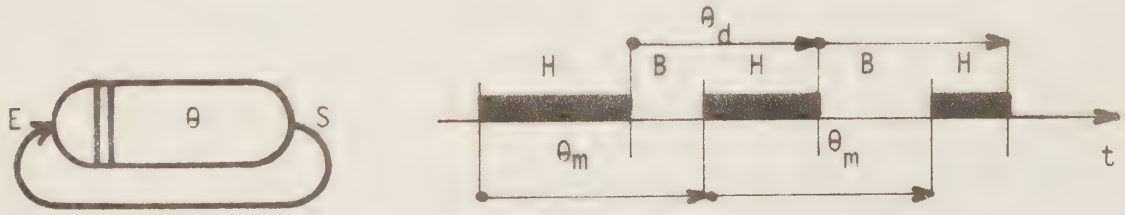
NOTE 1

LES CIRCUITS OSCILLATEURS

Un circuit oscillateur est constitué d'une chaîne bouclée de fonctions de retard. Un tel circuit permet de générer les signaux de référence, dont la durée des composantes est égale à celle d'un des temps de retard de la chaîne. Une telle génération des références permet d'évaluer la durée de leur composantes sans avoir à les comparer à d'autres signaux.

Conditions d'oscillation

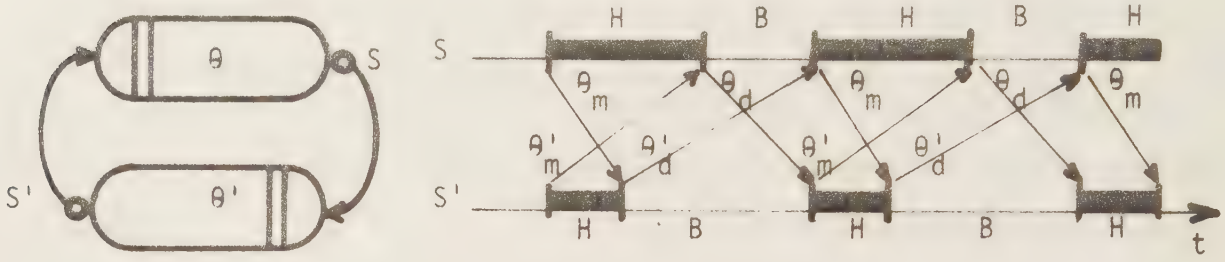
D'un point de vue théorique, le circuit oscillateur élémentaire peut être constitué d'une fonction de retard "bouclée sur elle-même" (logique à seuil unique, c'est à dire à alphabet binaire). Pour que ce circuit puisse osciller périodiquement, les temps de retard à la montée et à la descente doivent être égaux ; nous avons vu dans le 1er chapitre que les circuits physiques actuels n'ont pas cette propriété.



EXEMPLE : FILTRAGE PROGRESSIF DES COMPOSANTES HAUTES H

Considérons alors le circuit constitué de deux fonctions de retard "bouclées l'une sur l'autre". Si ces fonctions n'inversent pas, alors, quelquesoit la durée des composantes, le circuit fermé finira par atteindre un état stable. Cette stabilité est due à la différence des temps de retard à la montée et à la descente. Les composantes d'un des deux types {H,B } sont allongées, alors que les autres sont raccourcies ; par itération ces dernières seront donc filtrées.

Il n'en est pas tout à fait de même lorsque les fonctions de retard introduisent une inversion ; il existe alors un cas d'oscillation. Cependant l'incertitude sur les paramètres des fonctions de retard conduit physiquement à un filtrage des composantes d'un certain sens, ce qui entraîne la mise en un état stable (mémorisation).



CIRCUIT OSCILLANT CONSTITUE DE DEUX TEMPS DE RETARD

contraintes d'oscillations :

$$1)\quad d_H + d_B = \theta'm + \theta'd$$

$$2)\quad d'_H + d'_B = \theta m + \theta d$$

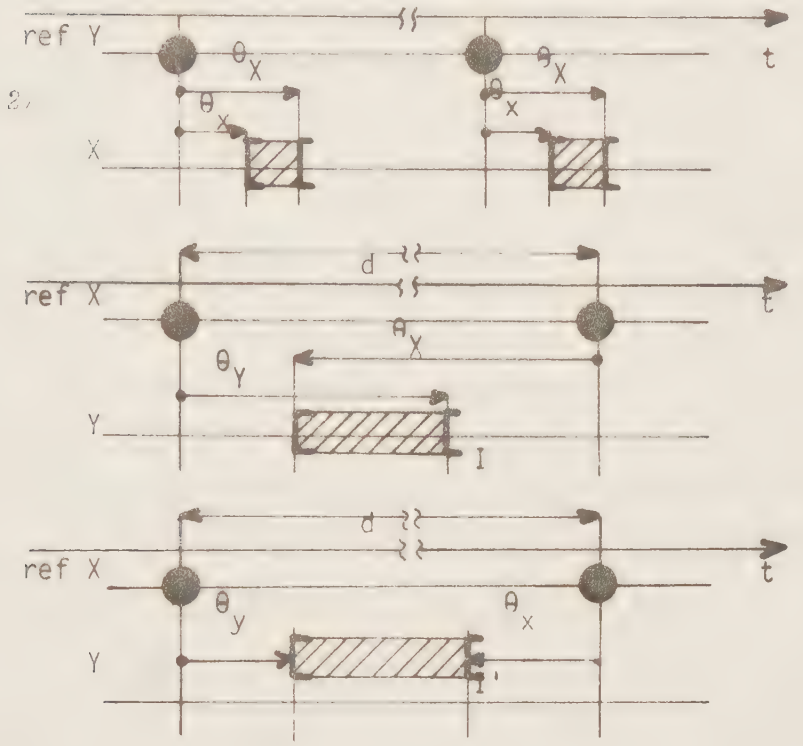
$$3)\quad d_H + d_B = d'_H + d'_B$$

Dans la pratique, les circuits oscillateurs sont constitués d'une boucle de deux fonctions de retard "à temps de relaxation" (éléments monostables) : les composantes d'entrée qui sont comprises entre
1) celle qui "déclenche" la relaxation et
2) la fin de la relaxation ou bien sont filtrées ou bien prolongent la durée de relaxation.
L'alphabet de sortie d'une telle fonction de retard est binaire et un des deux caractères définit l'état de relaxation (dont la durée est le "temps de retard" de la fonction).

Dans le cas d'un circuit oscillateur constitué de deux fonctions de retard (ou pseudo fonctions), nous avons un système de deux signaux qui sont mutuellement le "retardé" de l'autre ! Relativement à l'un d'entre eux, pris comme référentiel, les relations temporelles sont celles qui sont indiquées sur la figure.

Le transformé d'un point P_x existe, donc l'intervalle I (respI') existe. Nous vérifions bien la relation :

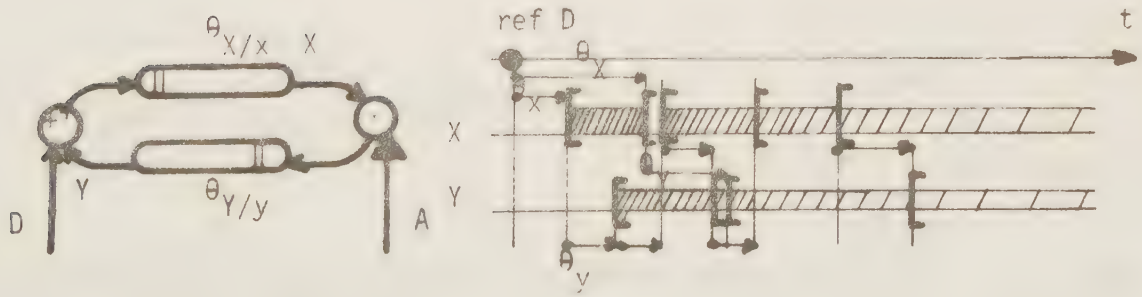
LOCALISATION DES TRANSFORMES DES SIGNAUX 1):X ET 2):Y



LOCALISATION DE L'ORIGINE DU SIGNAL X

RELATIONS TEMPORELLES ENTRE DEUX SIGNAUX D'HORLOGE X ET Y

CUMUL DES INCERTITUDES SUR LES TEMPS DE RETARD

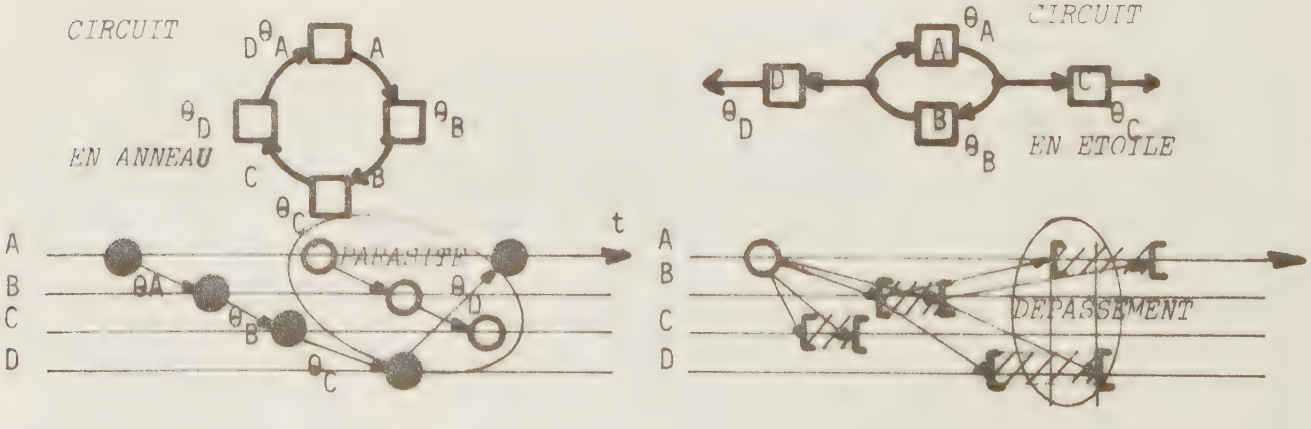


$$\theta x + \theta y \leqslant d \leqslant \theta x + \theta y$$

(le "transformé" Py peut être évalué de deux façons : 1) comme transformé de Px ou bien 2) comme "origine" de P'x).
Sur tous les diagrammes de temps précédents, nous avons supposé que le signal de référence était "synchronisé" sur le référentiel absolu en tout ses points. Lorsqu'il n'y a qu'un point de synchronisation, l'observation des signaux X et Y, relativement au référentiel absolu conduit à un cumul des incertitudes. Il en résulte que si le circuit oscillateur est "déclenché" par un troisième signal D, les signaux X et Y ne sont plus observables par rapport à celui-ci (D), au bout d'un certain nombre de périodes.

Le circuit fermé précédent possède la propriété de garantir la position mutuelle des "points" des deux signaux,quelle que soit la fluctuation temporelle des temps de retard. Chaque point d'un signal est en effet évalué par rapport à deux points qui l'"encadrent", de l'autre signal. Cette propriété est fondamentale pour les mécanismes périodiques, car elle seule permet d'affirmer que le dernier évènement d'une période précède le premier de la période suivante !

Il n'en est pas de même pour les circuits en "étoile", qui sont composés d'un circuit oscillateur qui "pilote" des chaînes de temps ; car pour une chaîne de temps, la position d'un "point" n'est évaluée que par rapport à un seul point du signal de référence. Il se peut alors qu'une variation (ie : retard dynamique) des temps de retard entraîne que ce point succède à son successeur théorique, relativement au référentiel absolu ! Les circuits en étoile sont cependant utilisés ; car dans les circuits fermés qui comportent plus de deux monostables, le déclenchement accidentel (parasite) de l'un d'eux peut doubler le nombre de points de référence !



CONCLUSION

En guise de conclusion, nous soulignons l'intérêt du modèle proposé dans les deux étapes de la conception d'un système séquentiel: la synthèse et l'analyse de vérification. Nous terminons en exposant les limites du modèle.

1. Analyse
2. Synthèse
3. Limites

CONCLUSION

Nous venons d'étudier un modèle temporel des systèmes séquentiels qui est fondé sur les principes de causalité, des effets retardés et de dualité des notions d'état et de temps. L'intérêt de ce modèle peut être évalué de deux façons, selon que son domaine d'application est l'analyse ou bien la synthèse d'un système séquentiel.

1. Analyse

L'analyse d'un système séquentiel est en grande partie temporelle. En particulier, cette analyse a pour but de déterminer les conditions temporelles pour lesquelles un système séquentiel vérifie sa fonction d'automate d'états.

Pour la théorie des *machines synchrones*, cette analyse est immédiate: elle consiste à évaluer les conditions de stabilisation des états du système. Malheureusement, la possibilité de cette évaluation n'est obtenue qu'au prix d'une limitation très stricte du domaine d'application de la théorie.

Pour la théorie des *machines asynchrones*, l'unicité du repérage ne permet pas d'effectuer complètement cette analyse et l'étude de ses "aléas" [UNG] ne peut conduire qu'à la délimitation du domaine d'application de la théorie. Un "aléa " correspond en effet à une "configuration" des signaux pour laquelle les relations de cause à effet ne peuvent être vérifiées,si tous les signaux sont repérés uniformément aux mêmes dates: *cela n'implique pas un "mauvais" fonctionnement du système séquentiel.*

A l'aide des *logiques booléennes temporelles*, il est par contre possible d'analyser la "réponse" d'un circuit séquentiel à un ensemble de signaux d'entrée donné. Cependant cette analyse ne permet pas de délimiter l'*ensemble* des solutions qui correspondent à la fonction d'automate d'états du système.

Les techniques actuelles de simulation des systèmes séquentiels sont fondées sur ce dernier type d'analyse. Le principe de base est l'"échantillonnage" de toutes les variables du système suivant un "pas" temporel suffisamment fin pour mettre en évidence les anomalies dues à des différences de temps de retard. Les techniques de simulation diffèrent principalement par la façon d'étudier les anomalies: - technique *parallèle* [SES],[HAR],[SZY] ou bien technique *déductive* [ARM], [CHA].

Pour notre part, nous ne pensons pas que la simulation permette de démontrer qu'un système séquentiel fonctionne "correctement" dans tous les cas. Nous nous limitons à considérer la simulation comme un moyen de"reproduire" un système, en général non réalisé physiquement. Les performances et la "fidélité" du simulateur dépendent alors du modèle de simulation choisi [NOG].

Le modèle temporel que nous proposons permet d'effectuer un type d'analyse plus général. Ce modèle permet en effet de *calculer les conditions limites* de fonctionnement temporel d'un système séquentiel défini comme un automate d'états. Le calcul de ces conditions peut tenir compte des incertitudes sur la valeur des temps de retard, quels que soient l'étendue du système et le cumul des incertitudes qui en résulte.

L'analyse est fondée sur le principe d'un repérage des états relatif à chaque signal. Elle consiste à déduire les caractéristiques temporelles d'un état du système à partir de l'état des signaux et des relations qui lient les signaux entre eux.

Le fait de dissocier les notions d'état et de temps supprime la relation d'*unicité* qui existe,dans les autres théories,entre un état et son repérage temporel. A un seul et même état correspond alors une infinité non dénombrable de systèmes de repérage.

Afin d'obtenir un ensemble "continu" de systèmes de repérage qui ne diffèrent que par la date et la durée d'observation, nous avons postulé que la *durée* d'un état élémentaire ne pouvait avoir qu'*une seule cause temporelle*. Cette hypothèse permet d'éliminer un certain nombre d'ensembles de systèmes de repérage mais elle ne garantit pas l'unicité de la solution. Seule la *définition* univoque de la séquence d'états stables,dans laquelle s'insère l'état étudié ,

permet de ne retenir qu'un seul ensemble continu de systèmes de repérage. Il y a en effet autant de solutions que de séquences d'états stables possibles, alors qu'aucune propriété ne distingue a priori un état stable d'un état transitoire: un état peut être considéré comme"stable"dès lors que la durée de sa cause de maintien est un paramètre du système de repérage (du moins pour les systèmes "dynamiques").

La méthode d'analyse que nous proposons ne peut donc être menée à son terme que si la séquence des états "stables" du système est définie, c'est à dire si la synthèse du système physique est complète. Une méthode "conversationnelle"s'avère nécessaire car,dans la pratique, la synthèse initiale est toujours partielle. Ce type de méthode permet d'éliminer au fur et à mesure un certain nombre d'états prédécesseurs de l'état étudié [*la méthode proposée détecte tous les états prédécesseurs possibles; leur nombre est fini car le nombre d'états de chaque signal est fini*].

D'après nos recherches bibliographiques, il semble bien qu'il n'y ait pas d'autre *méthode* d'analyse qui soit basée sur les mêmes fondements. Nous ne connaissons que deux analyses "pragmatiques" de ce genre de problème:

- celle que D. Etiemble a effectuée à l'aide de son modèle des éléments logiques [ETI]
- et une autre que nous contestons [SER]: nous avons remarqué en particulier que les contraintes temporelles sur les entrées *symétriques* d'un élément logique (du système d'échantillonnage) étaient différentes!

L'évaluation des coordonnées temporelles limites d'un état a été effectuée en supposant que les temps de retard " à la montée " et " à la descente " des signaux étaient identiques. Cette hypothèse a été introduite dans le seul but de simplifier l'exposé de la méthode d'analyse. L'introduction d'une dissymétrie des temps de retard n'a en fait véritablement de sens que si les durées minimales des composantes - durée en dessous de laquelle les composantes sont filtrées - sont prises en compte. Les contraintes temporelles ont alors une expression plus complexe qui ne facilite pas la lecture (l'analyse de la bascule RS a été effectuée initialement avec ces hypothèses).

2. Synthèse

Dans le modèle proposé, les étapes de la synthèse d'un système séquentiel sont identiques à celles des autres théories:
- 1) synthèse d'un système à l'aide de cellules de mémoire;
- 2) synthèse d'une cellule de mémoire à l'aide d'éléments combinatoires.

Si les étapes de la synthèse sont identiques à celles des autres théories, les principes sont par contre différents.

Dans la première étape, notre point de vue est celui de la *microprogrammation*, point de vue que nous avons exposé par ailleurs [NOG]. En général, la définition du nombre d'éléments de mémoire ne résulte pas de l'étude des états du système séquentiel, mais résulte d'impératifs techniques. La synthèse se fait alors en termes de transferts et l'"optimisation" de la structure provient avant tout de la *qualité de l'énoncé*.

Ce genre de problème relève surtout d'une méthodologie comparable à la *programmation structurée* [ARS]. Des recherches antérieures nous ont persuadé que cette méthodologie peut s'appliquer à la description d'une machine et même à sa conception [NOG]. Notons qu'il en est de même pour la théorie des machines synchrones [HYD].

La démarche que nous empruntons dans cette première étape de la synthèse pourrait être comparée à celle qui a été prise pour le système ALERT [FRI]. La particularité de notre méthode est de poser formellement le problème d' "optimisation" des commandes en termes de *transparence* des bascules. En fait, cette démarche est adoptée, de façon informelle, par ceux qui utilisent les éléments logiques d'un "catalogue".

Pour la deuxième étape de la synthèse, la méthode utilisée est spécifique à notre modèle temporel: les problèmes sont posés en termes "logiques" d'*établissement* et de *maintien* d'un état. Cette approche permet d'effectuer la synthèse des différents types de bascules existants. Cette synthèse est indépendante de la technologie: elle est purement logique, ce qui est justifié puisque le but est de définir un système équivalent à un automate abstrait d'états. Nos conclusions s'opposent, pour certaines propriétés des bascules *"Maître - Esclave"*, à celles qui sont faites dans d'autres ouvrages.

Pour conclure cette partie, le modèle temporel proposé permet d'analyser les caractéristiques temporelles des systèmes séquentiels et cela,à partir d'une définition *abstraite* d'automate d'états. Les exemples fournis permettent de vérifier la validité de cette analyse; ils correspondent à des cas concrets, physiquement réalisés. Les algorithmes fournis sont fondés sur des méthodes d'analyse bien connues en compilation; c'est pourquoi nous pensons que l' "automatisation" du modèle proposé ne doit pas poser de gros problèmes. Une implantation sur ordinateur, en mode "conversationnel", est actuellement à l'étude.

3. Limites du modèle

Si notre modèle permet de calculer les conditions limites de fonctionnement, il ne permet pas pour autant d'étudier la propagation des *bruits* à l'intérieur d'un système séquentiel. La raison en est simple: les caractéristiques temporelles d'un bruit sont aléatoires, on ne peut donc traiter ce bruit comme une composante de*"commutation"* dont les caractéristiques temporelles peuvent se déduire de celles des éléments logiques. Ce type de phénomène sort ainsi du domaine d'application du modèle temporel.

L'étude de la propagation d'un bruit nécessiterait une représentation plus complexe des composantes de signaux. Ainsi, le doublet < *état, durée* > devrait être remplacé au moins par un triplet < *temps de transition, état, durée* >. De plus, les entrées d*"alimentation"* devraient être incluses dans le modèle d'un élément logique car, d'une part ces entrées servent de *référence d'état* (dans notre modèle nous avons considéré les *références de temps* - eg: les signaux d'horloge - *comme des entrées normales*) et, d'autre part, il est bien connu que la grande majorité des bruits sont propagés par "l'alimentation".

En plus de cette limite "vers le bas", le modèle proposé se heurte, à l'autre bout de l'échelle de classification temporelle des systèmes séquentiels, aux problèmes de synchronisation. A ce niveau, la notion de temps "continu" se transforme en celle de séquence d'évènements. Cette transformation est difficile à localiser dans une structure hiérarchique [ANC] des systèmes. La structure physique d'une machine est toujours "optimisée", c'est pourquoi il n'existe pas de frontière temporelle bien nette, telle que pour les éléments inférieurs de la hiérarchie le temps soit "continu", alors qu'il prendrait une forme "discrète" pour les éléments supérieurs.

Par exemple, les problèmes temporels que posent les transferts à l'intérieur d'une machine n'apparaissent pas au niveau du"langage-machine", par contre ceux que pose la gestion des périphériques apparaissent sous la forme d'instructions de test et d'attente: la mesure de la durée "réelle" des échanges est alors définie à ce niveau.

L'approche prise pour la décomposition des procédures d'échanges en un emboîtement de langages, dont certains ne concernent que la synchronisation des organes émetteurs et récepteurs, nous paraît intéressante pour formaliser ce problème [SNA],[DRO].

ANNEXE

Afin de ne pas alourdir l'étude du modèle temporel, nous présentons sous forme d'annexe l'analyse des propriétés temporelles des systèmes combinatoires. Nous étudions d'abord les problèmes de repèrage que pose la fonction de retard. Nous proposons ensuite une classification des éléments combinatoires relative aux caractéristiques temporelles de leur fonction logique.

APPLICATION DU MODELE
AUX SYSTEMES COMBINATOIRES

I - Le retard simple

II - Les fonctions combinatoires

MODELE SYNTHETIQUE DES ELEMENTS LOGIQUES

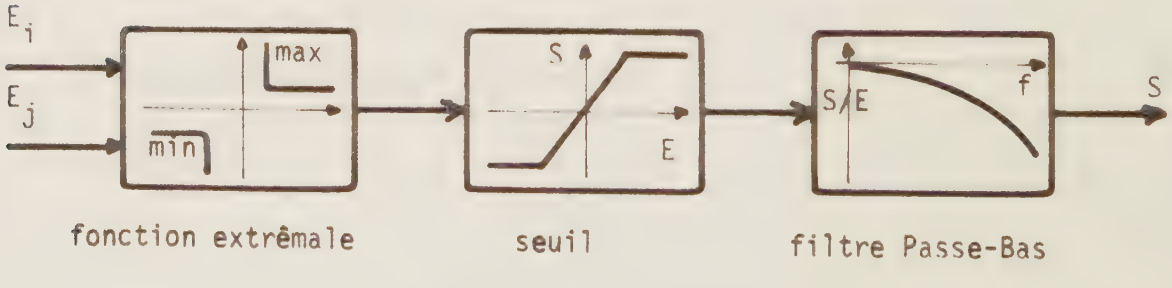
L'étude des caractéristiques temporelles d'un ensemble d'éléments logiques - c'est à dire d'une machine - nécessite un modèle synthétique et général des éléments. Les composantes de ce modèle doivent mettre en évidence les différentes propriétés physiques des signaux impliquées dans le traitement logique de l'information.:

- *combinaison des signaux,*
- *seuil,*
- *temps.*

Il en est ainsi du modèle proposé par J. Suchard [*SUC*]. *Bien que simple, ce modèle permet d'évaluer précisément les propriétés dynamiques d'un élément logique. La validité de ce modèle a été démontrée et vérifiée par les mesures, pour la logique TTL bas niveau, par D. Etiemble* [*ETI*].

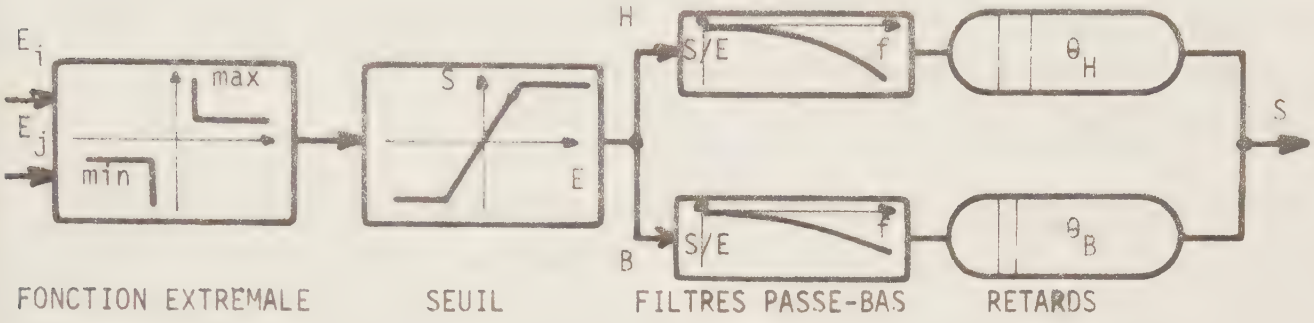
Le modèle de base de J. Suchard décompose un élément logique en trois types de fonction :

1) les fonctions combinatoires continues et extrêmales, indépendantes du temps,
2) les seuils,
3) les filtres linéaires "passe-bas".



MODELE DE J. SUCHARD

Dans un modèle dérivé, D. Etiemble a dédoublé le filtre de sortie pour tenir compte des propriétés dissymétriques de la logique de commutation de courant; l'addition de retards purs s'est alors avérée nécessaire. Notons que la grandeur caractéristique du modèle est la tension.



MODELE DE D. ETIEMBLE

LES PARAMETRES TEMPORELS

Comme nous l'avons déjà dit, les seuls points repèrables d'un signal sont les points de transition entre deux composantes distinctes. Relativement à la fonction de seuil, un point de transition est défini par l'égalité de la valeur des signaux d'entrée et de sortie. Cette tension de seuil est la "coupure" L entre les deux domaines de variation V_0 et V_1 qui caractérisent l'alphabet binaire associé au signal. Les points de transition délimitent donc les "caractères".

Comme les fonctions extrêmales sont indépendantes du temps - dans les modèles de J. Suchard et D. Etiemble - les points limites des caractères à la sortie de ces fonctions ont une date identique à celle de certains points de transition des signaux d'entrée. D' un point de vue temporel, les seuls paramètres importants sont alors les temps de retard de transmission des points de transition, dûs aux fonctions de seuil et de filtre.

Ces temps de retard dépendent, dans les modèles précédents, des temps de montée (ou de descente) du signal, mesurés aux points de transition. Cette dépendance implique théoriquement l'adjonction d'un paramètre supplémentaire pour caractériser un point de transition. Chaque point de transition devrait alors être caractérisé par un triplé < date, sens de la transition, temps de montée/descente >.

En fait, D. Etiemble a montré (pour les modèles précédents) que les temps de montée/descente des signaux convergent très rapidement vers une valeur qui ne dépend que des caractéristiques des éléments logiques traversés (ie: du filtre et du seuil). Nous pouvons donc admettre que ces temps de montée/descente sont les mêmes pour tous les signaux si les éléments logiques ont des caractéristiques physiques voisines (ce que nous supposerons). Il en résulte que le paramètre "temps de transition" n'est pas spécifique à un point de transition particulier.

Lorsque les points de transition sont suffisamment éloignés temporellement les uns des autres, il y a une correspondance biunivoque entre les valeurs des temps de montée et de descente et les temps de retard correspondants. Dans ce cas, les temps de montée/descente peuvent être exclus du raisonnement qui permet de déterminer le transformé d'un signal après la traversée d'un élément logique.

Lorsque l'intervalle de temps qui sépare deux points de transition tend vers zéro, les temps de montée/descente influent alors sur le raisonnement. La connaissance des temps de transition et de la durée de l'intervalle de temps permet en effet de déterminer si le signal "a le temps" d'atteindre sa valeur nominale (eg: 0_V ou 5_V) entre les deux points de transition.

Schématiquement, le fait d'atteindre ou non la valeur nominale délimite deux modes de fonctionnement:

- *"stable" (la valeur nominale est alors atteinte),*
- *"instable".*

Pour le premier mode, les propriétés de convergence des temps de transition sont vérifiées. Par contre, pour l'autre mode de fonctionnement, D. Etiemble a mis en évidence un phénomène d' "avalanche": par traversée successive d'éléments logiques "inverseurs" la valeur extrêmale de la composante de signal s'écarte de plus en plus de la valeur nominale. Autrement dit, cette composante est "filtrée" progressivement.

Cependant, les propriétés des modèles proposés permettent de définir ce phénomène d'"avalanche" uniquement en termes de dates de transition.

1) Pour celà, nous devons postuler que pour toute combinaison logique, les signaux ont des temps de transition dont la valeur dépend uniquement des caractéristiques des éléments logiques. Ceci implique que tous les signaux

d'entrée sont "mis en forme" avant d'être traités, c'est à dire que leur temps de transition ont atteint la valeur de convergence. Dans la pratique, nous pouvons admettre que cette mise en forme est réalisée après la traversée des éléments d'"adaptation", car la convergence est extrêmement forte.

2) Dans ces conditions, l'amplitude du phénomène d'avalanche peut être exprimée uniquement en fonction de la durée initiale d'une composante instable. Celà résulte du fait a) que cette composante ne peut avoir pour origine que la combinaison de deux composantes "stables" (hypothèse 1) et b) que la durée de la composante peut être déduite des dates des points de transition des signaux "stables" combinés, puisque le modèle suppose que la combinaison est "intemporelle".

Il résulte de ces différentes remarques que les temps de transition peuvent dans tous les cas être implicites: la date des points de transition permet de spécifier complètement les phénomènes. Celà implique naturellement que les paramètres des éléments logiques, caractéristiques du modèle, sont connus implicitement afin que l'on puisse définir quantativement les phénomènes.

Pour la logique "à commutation de courant", la dissymétrie de fonctionnement entraîne une différence entre les temps de retard 1) des transitions "montantes" et 2)"descendantes". Cette dissymétrie influe sur le filtrage des composantes dont la durée n'est pas suffisante. A l'aide de son modèle, D. Etiemble [§IV,1:8] a calculé pour les deux types de composantes (V_0 et V_1) la durée minimum en dessous de laquelle l'amplitude extrêmale du transformé de la composante:

- 1) n'atteint pas la valeur nominale, mais franchit cependant la tension de seuil (mode de fonctionnement "instable"), cette durée sera notée d_S;
- 2) ne franchit pas la tension de seuil.

Dans le second cas, la composante est purement "éliminée" par définition, puisque les transformés des points de transition n'existent pas.

Dans le premier cas, il y a -théoriquement- une composante de sortie, puisqu'elle "franchit" la tension de seuil. Cependant, le gain dynamique est tellement élevé au voisinage de la tension de seuil, que la moindre variation temporelle des caractéristiques élimine ou au contraire "remet en forme" cette composante.

Nous supposerons dans notre raisonnement que toute composante a une durée supérieure à la borne critique (d_S). Car notre but est de déterminer les relations temporelles, entre les repères de signaux, qui garantissent de façon certaine le fonctionnement logique. Nous rappelons à ce propos, que la notion d'aléa n'est pas relative au signal physique, mais dépend uniquement du repèrage (logique) du signal.

I) LE RETARD SIMPLE

Nous étudions dans cette partie les relations temporelles qui lient deux signaux dont l'un est le "retardé" de l'autre. Nous nous limitons à l'étude de la "logique à seuil unique", c'est à dire à l'étude des signaux dont l'"alphabet" associé ne comprend que deux classes de composantes : "hautes" et "basses".

1.1. Caractéristiques physiques

Nous venons de voir que la fonction de retard pouvait être caractérisée à l'aide des paramètres suivants, dans le cas d'un logique binaire :

Temps de retard : - à la "montée" (θ_m),
- à la "descente (θ_d),

Durée minimale d'une composante : - "haute" (d_h),
- "basse" (d_b).

Les temps de retard peuvent être statiques, c'est à dire invariants et indépendants de l'environnement "dynamique" (par exemple : le temps de montée des signaux, le nombre d'éléments couplés à la sortie qui débitent ou absorbent du courant à un instant donné etc...) Dans le cas contraire, les temps de retard sont dits "dynamiques".

Lorsque les composantes d'entrée ont une durée inférieure à la durée minimale, leur transmission est aléatoire, si ce n'est impossible : elles sont alors filtrées.

Ces quatres paramètres sont mesurés avec une certaine approximation (ie : incertitude). Nous noterons (θ_m, θ_d, d_h, d_b) leur valeur minimale et (θ_M, θ_D, d_H, d_B) leur valeur maximale.

1.2. Repérage

Nous notons E le signal d'entrée et R_E son repère. Le signal de sortie - c'est à dire le signal retardé - est noté S et R_S son repère. Chacun des deux signaux E et S peut être choisi comme référence ; l'autre signal est alors "observé" par rapport à cette référence.

Le repère du signal de référence est supposé daté sur le référentiel "absolu". C'est à dire qu'à chaque borne d'une composante de ce repère est supposée associée une date (ie : un point) du référentiel. Le transformé - par la fonction de retard - de cette borne est évalué sur le même référentiel. A cause des incertitudes, ce transformé est un intervalle, dont les bornes délimitent les composantes de l'autre repère. La position et la durée de ces composantes en sont déduites.

Les relations temporelles ainsi obtenues sont donc :
1) évaluées sur l'échelle de mesure du référentiel "absolu",
2) relatives au signal de référence. De plus, elles dépendent du type partiel ou complet des repères.

1.2.1. REPERE COMPLET

Par définition, toutes les composantes d'un signal, repéré complètement, sont des "caractères" de l'alphabet {H,B }. Cet alphabet correspond à une partition de l'ensemble physique des composantes. Relativement à l'utilisation logique d'un signal, nous avons distingué les composantes caractéristiques (A) et les composantes neutres (e). La correspondance, entre cet alphabet logique {A,e } et l'alphabet physique {H,B }, dépend des fonctions extrêmales utilisées. Pour les fonctions maximales, A correspond à H et e à B . Par contre, pour les fonctions minimales A correspond à B et e à H . Il en résulte que la valeur des paramètres de la fonction de retard, exprimée par rapport à l'alphabet logique, dépend des conventions choisies :

fonctions maximales : $\theta_a = \theta_m$, $\theta_e = \theta_d$, $d_a = d_h$, $d_e = d_b$;
$\theta_A = \theta_M$, $\theta_E = \theta_D$, $d_A = d_H$, $d_E = d_B$.

fonctions minimales : $\theta_a = \theta_d$, $\theta_e = \theta_m$, $d_a = d_b$, $d_e = d_h$;
$\theta_A = \theta_D$, $\theta_E = \theta_M$, $d_A = d_B$, $d_E = d_H$.

1.2.2. REPERE PARTIEL

La classification logique des composantes ne correspond plus à leur valeur, mais à la probabilité égale à un d'avoir uniquement une des deux valeurs possibles. Il en résulte qu'une composante définie (A) peut être :

1) - soit une composante "haute" H,
2) - soit une composante "basse" B,
3) - soit ou bien une composante H ou bien une composante B (ou exclusif).

Relativement à l'alphabet logique (A,e) les paramètres de la fonction de retard prennent des valeurs différentes dans chacun des trois cas. Nous avons en particulier les relations suivantes.

1) $\theta_a = \theta_m$, $\theta_E = \theta_D$, $\theta_A = \theta_M$, $\theta_e = \theta_d$, $d_a = d_h$, $d_A = d_H$

2) $\theta_a = \theta_d$, $\theta_E = \theta_M$, $\theta_A = \theta_D$, $\theta_e = \theta_m$, $d_a = d_b$, $d_A = d_B$

3) $\theta_a = \min(\theta_m, \theta_d)$, $\theta_E = \max(\theta_M, \theta_D)$, $\theta_A = \max(\theta_M, \theta_D)$, $\theta_e = \min(\theta_m, \theta_d)$, $d_a = \min(d_b, d_h)$, $d_A = \max(d_B, d_H)$

1.3. Composantes parasites

Les composantes dont la durée est inférieure à la borne "critique" sont filtrées. Nous n'en tiendrons donc pas compte dans le raisonnement. Cependant, lorsque le signal d'entrée comporte de telles composantes, celles-ci ne doivent pas altérer le signal de sortie.

L'altération proviendrait du fait que l'insertion d'une composante "parasite" (p), dans une composante caractéristique d'entrée (C_e), décompose celle-ci en une concaténation ($x^{\frown} p^{\frown} y$) dont la durée des composantes x et y serait inférieure à la borne critique. Une composante parasite filtrable a donc les caractéristiques suivantes :

1) $d(x^{\frown} p^{\frown} y) = d(C_e)$,
2) $d(x) \geqslant d_A$ et $d(y) \geqslant d_A$,
3) $d(p) \leqslant d_a$,
4) x, y et Ce appartiennent à la même partie de la partition {H,B } des composantes.

Une composante parasite filtrable ne peut être un "bruit aléatoire", dont - par définition - la position temporelle est incertaine. Par contre, ce type de composante peut être le résultat d'une combinaison "limite" de signaux. Un tel "aléas" peut être admis, s'il vérifie les conditions précédentes, car il est filtré.

Remarquons d'autre part qu'un repère d'entrée, de type partiel et dont la durée des composantes indéfinies est inférieure à la borne critique, donne en sortie un repère de type complet, puisque toutes les composantes indéfinies sont filtrées. Il s'agit d'un cas vraiment très particulier.

1.4. But recherché

Les relations temporelles qui lient les repères d'entrée et de sortie dépendent de ce que l'on cherche à évaluer. Deux points de vue sont envisageables :

1) le point de vue de l'"observation" : il s'agit de déterminer, relativement au référentiel, les parties de l'autre signal qui ont une possibilité égale à 1 d'être un "caractère" de l'alphabet. Leur détermination est nécessaire pour définir toute combinaison de signaux.
2) le point de vue de la "conception" : il s'agit alors de déterminer les caractéristiques minimales du signal d'entrée (la cause) qui garantissent dans tous les cas les caractéristiques imposées du signal de sortie (l'effet).

1.5. Relations temporelles

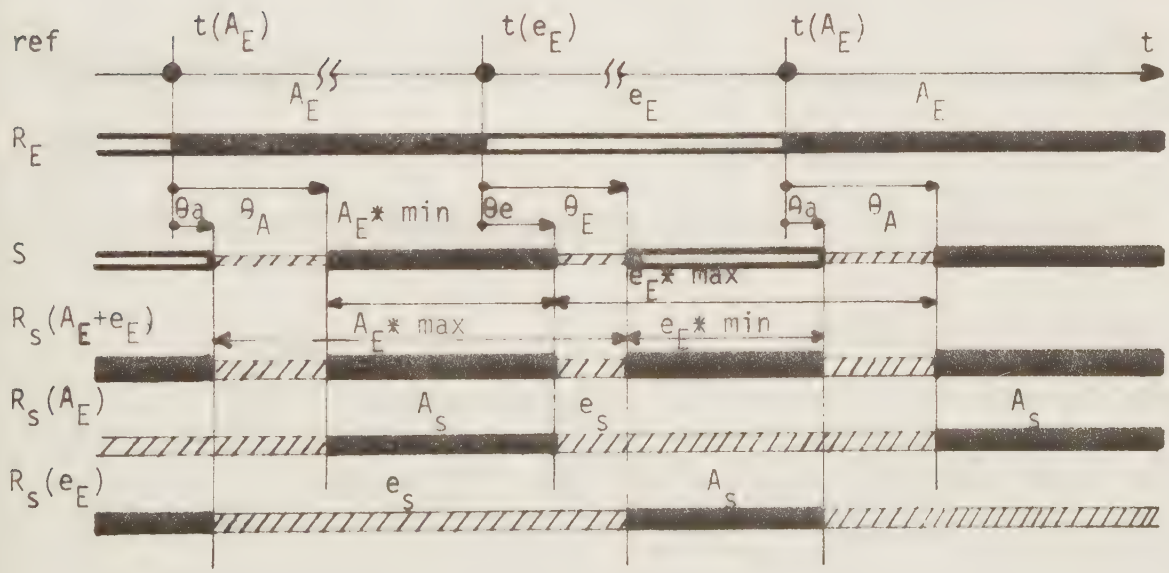
L'expression des relations temporelles dépend de nombreux facteurs :

- 1) choix de la référence : entrée ou sortie,
- 2) type du repère de référence : partiel ou complet,
- 3) choix du repère partiel "observé" : il y a 3 possibilités pour les composantes définies,
- 4) but recherché : observation ou conception.

Nous étudierons successivement tous les différents cas possibles. Nous n'effectuerons plus par la suite ce genre d'étude, c'est la raison pour laquelle nous entrons au moins une fois dans les détails.

1.5.1. OBSERVATION DES CARACTERISTIQUES PAR RAPPORT A UN REFERENTIEL

1.5.1.1. Repère complet

1.5.1.1.1. Référence d'entrée

a) caractéristiques minimales

durée: $d(A_E\star) = d(A_E) + \theta_e - \theta_A$; $d(e_E\star) = d(e_E) + \theta_a - \theta_E$

retard: $t(A_E\star) = t(A_E) + \theta_A$; $t(e_E\star) = t(e_E) + \theta_E$

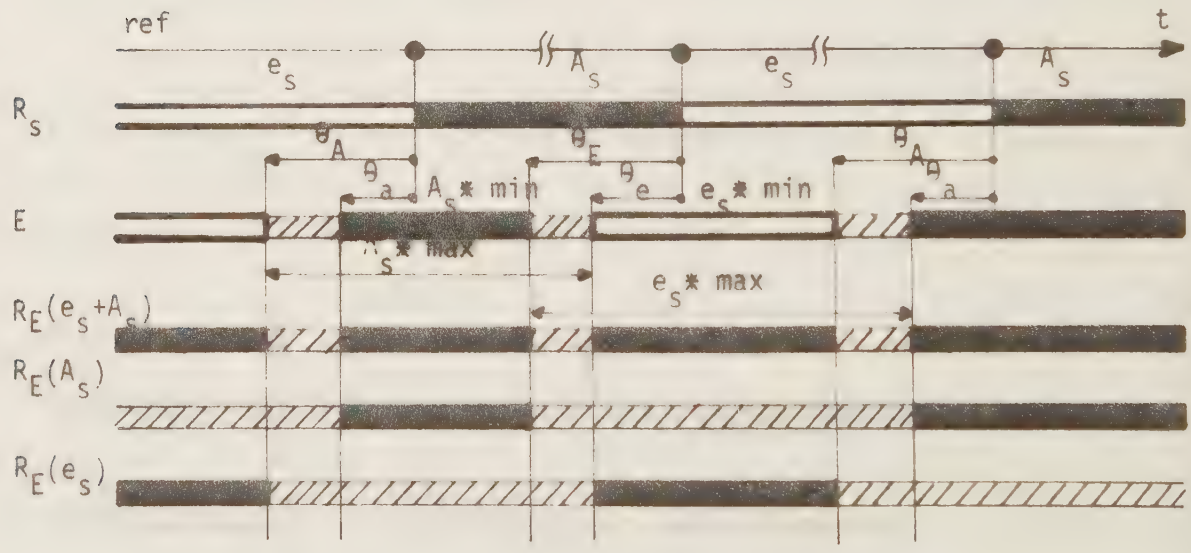
b) caractéristiques maximales

durée: $d(A_E\star) = d(A_E) + \theta_E - \theta_a$; $d(e_E\star) = d(e_E) + \theta_A - \theta_e$

retard: $t(A_E\star) = t(A_E) + \theta_a$; $t(e_E\star) = t(e_E) + \theta_e$

Trois repères de type partiel [$R_S(A_E+e_E)$, $R_S(A_E)$, $R_S(e_E)$] peuvent être associés au signal de sortie. Ces trois repères n'existent que si la durée minimum des composantes A_E et e_E est supérieure à la valeur critique (d_A ou d_E).

1.5.1.1.2 Référence de sortie



a) caractéristiques minimales

durée: $d(A_S\star) = d(A_S) + \theta_a - \theta_E$; $d(e_S\star) = d(e_S) + \theta_e - \theta_A$

avance: $t(A_S\star) = t(A_S) - \theta_a$; $t(e_S\star) = t(e_S) - \theta_e$

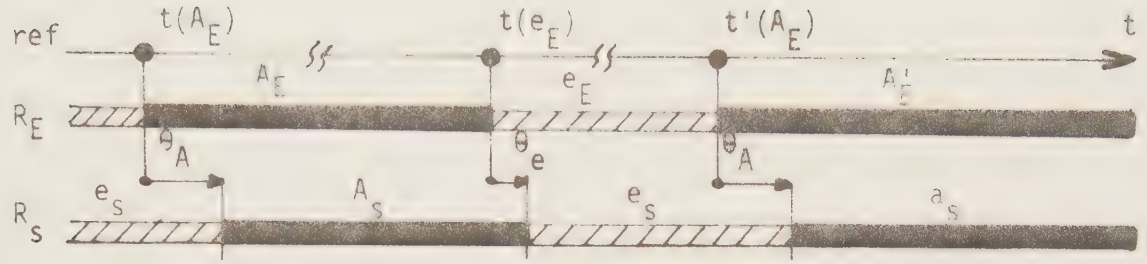
b) caractéristiques maximales

durée: $d(A_S\star) = d(A_S) + \theta_A - \theta_e$; $d(e_S\star) = d(e_S) + \theta_E - \theta_a$

avance: $t(A_S\star) = t(A_S) - \theta_A$; $t(e_S\star) = t(e_S) - \theta_E$

1.5.1.2 Repère partiel

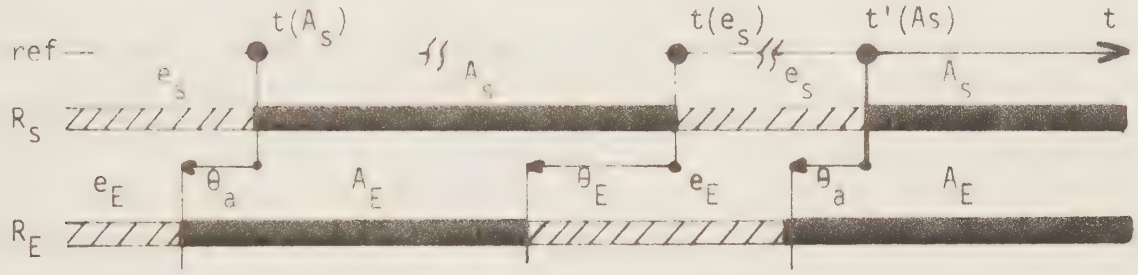
1.5.1.2.1 Référence d'entrée



durée: $d(A_S) = d(A_E) + \theta_e - \theta_A$; $d(e_S) = d(e_E) + \theta_A - \theta_e$

retard: $t(A_S) = t(A_E) + \theta_A$; $t(e_S) = t(e_E) + \theta_e$

1.5.1.2.2 Référence de sortie



durée: $d(A_E) = d(A_S) + \theta_a - \theta_E$; $d(e_E) = d(e_S) + \theta_E - \theta_a$

avance: $t(A_E) = t(A_S) - \theta_a$; $t(e_E) = t(e_S) - \theta_E$

1.5.2 DETERMINATION DES CARACTERISTIQUES MINIMALES

1.5.2.1 Repère partiel de sortie

Les caractéristiques sont obtenues en prenant le repère d'entrée comme référence, puisque le signal d'entrée est la cause du signal de sortie (effet). Relativement aux caractéristiques minimales <u>imposées</u> du signal de sortie [$d(A_S)$, $d(e_S)$, $t(A_S)$, $t(e_S)$], les caractéristiques minimales du signal d'entrée sont alors les suivantes.

$$d(A_E) = d(A_S) + \theta_A - \theta_e$$
$$d(e_E) = d(e_S) + \theta_e - \theta_A$$
$$t(A_E) = t(A_S) - \theta_A$$
$$t(e_E) = t(e_S) - \theta_e$$

Ces caractéristiques sont différentes de celles qui peuvent être "observées" à partir du signal de sortie pris comme référence.

1.5.2.2. Repère complet de sortie

Lorsque le repère de sortie est de type complet cela signifie que le signal de sortie est utilisé comme référentiel, par rapport aux autres signaux.

A cause de l'incertitude sur la valeur des paramètres de la fonction de retard, il est impossible de déterminer un signal d'entrée dont les caractéristiques garantissent celles du signal de sortie de façon certaine. L'introduction de contraintes sur les caractéristiques d'entrée permet seulement de garantir des bornes inférieures et supérieures à la durée des composantes de la référence de sortie. Ces bornes correspondent aux caractéristiques minimales et maximales qui ont été évaluées par rapport au signal d'entrée (1.5.1.1.1).

Si $[d(A_s), d(e_s), t(A_s), t(e_s)]$ et $[D(A_s), D(e_s), T(A_s), T(e_s)]$ sont les bornes extrêmes imposées aux composantes de sortie (par rapport au référentiel "absolu"), le signal d'entrée doit avoir les caractéristiques suivantes.

caractéristiques minimales :

$$d(A_E) \geq d(A_s) + \theta_A - \theta_e \; ; \; d(e_E) \geq d(e_s) + \theta_E - \theta_a$$
$$t(A_E) \geq t(A_s) - \theta_a \quad ; \; t(e_E) \geq t(e_s) - \theta_e$$

caractéristiques maximales

$$D(A_E) \leq D(A_s) + \theta_a - \theta_E \; ; \; D(e_s) \leq D(e_s) + \theta_e - \theta_A$$
$$T(A_E) \leq T(A_s) - \theta_A \quad ; \; T(e_E) \leq T(e_s) - \theta_E$$

1.6. Cumul des temps de retard

Par composition (ie : par "mise en chaîne"), la fonction de retard donne une fonction de même type. L'évaluation de ses paramètres doit être faite par rapport aux alphabets logiques, associés aux signaux de la "chaîne" et non pas par rapport à l'alphabet "physique" {H,B }. Car il peut y avoir des inversions - par rapport à l'alphabet { H,B} - qui n'ont aucune signification logique (mais qui influent cependant sur la valeur des paramètres globaux de la "chaîne de retards".

Si la chaîne de retards comprend n éléments, nous obtenons les mêmes relations temporelles que pour une fonction de retard unique, si les conditions suivantes sont vérifiées :

1) la durée de chacune des composantes, du signal d'entrée et des signaux intermédiaires, est supérieure à la borne critique de filtrage ;
2) les temps de retard ont pour valeur :

$$\theta_a = \sum_{i=1}^{n} \theta_{a_i} \qquad \text{et} \qquad \theta_e = \sum_{i=1}^{n} \theta_{e_i}$$

$$\theta_A = \sum_{i=1}^{n} \theta_{Ai} \qquad\qquad \theta_E = \sum_{i=1}^{n} \theta_{Ei}$$

Les inversions "physiques" permettent de compenser la différence entre les temps de retard $\theta(A)$ et $\theta(e)$. En effet, si l'on suppose 1) que tous les éléments de la chaîne ont les mêmes caractéristiques et 2) qu'une inversion "raccourcit" la durée d'une composante de la différence $|\theta(A) - \theta(e)|$, alors l'inversion suivante "rallonge" cette durée de la même valeur $|\theta(A) - \theta(e)|$. Il en résulte que, pour un nombre pair d'inversions, la durée des composantes est à peu près invariante et risque moins d'être filtrée, à cause d'un "raccourcissement" dû à la différence des temps de retard.

1.7. Distance invariante

Les caractéristiques temporelles précédentes sont déterminées à l'aide de relations ponctuelles. Le transformé d'un signal est ainsi défini par les transformés des points de transition entre deux

composantes distinctes ; car ce sont les seuls points discernables.
La différence des temps de retard des transitions "montantes" et "descendantes" entraîne que la durée d'une composante n'est pas invariante. Il n'en est pas de même, si les durées sont définies entre deux points de transition de même sens. Ces deux points sont "retardés" de la même façon, mais pas obligatoirement de la même valeur ; celà dépend de la nature des retards. Lorsque les retards sont "statiques", les valeurs sont les mêmes et sont "entâchées" de la même incertitude. Dans ce cas, la distance définie par ces deux points est invariante. Par contre, lorsque les retards sont "dynamiques", les deux valeurs sont indépendantes. La distance définie par ces deux points de transition peut alors varier, mais sa variation (v) est inférieure à deux fois l'incertitude sur le temps de retard:

$$0 \leqslant v \leqslant 2(\theta_M - \theta_m)$$

ou bien: $$0 \leqslant v \leqslant 2(\theta_D - \theta_d)$$

II - LES FONCTIONS COMBINATOIRES

Nous pouvons déduire de l'étude du retard simple que la notion de simultanéité (ie: de point) est une notion relative au référentiel choisi et qui est liée aux relations d'incertitude. C'est pourquoi nous éviterons de définir une fonction combinatoire en des termes qui impliquent cette notion de point. Nous rejetons ainsi les définitions du genre suivant :

1) (point de vue "logique") : "une fonction combinatoire combine plusieurs informations (ou caractères) pour en fournir une autre" ;
2) (point de vue "physique"): "une fonction combinatoire combine plusieurs signaux pour en fournir un autre".

Nous sommes d'autant plus motivés que d'autres définitions "possibles" excluent la notion de simultanéité ; en particulier lorsque les signaux sont considérés comme des suites. Deux suites peuvent en effet être "combinées" de différentes façons ; par exemple :

- 1) en prenant une composante de l'une ou sinon de l'autre (mutuelle exclusion),
- 2) en intercalant les composantes des deux suites (union)
- 3) en réunissant une composante de chaque suite afin de n'en obtenir qu'une seule (intersection).

Remarquons que la notion de simultanéité peut être retrouvée, si l'on introduit la notion de séparateur (ie : d'élément neutre).

Pour éviter cet écueil, nous introduirons les fonctions combinatoires en d'autres termes. Pour celà, nous considérons un signal comme l'ensemble des suites possibles d'évènements. Une fonction combinatoire est alors une application (C) d'un produit de signaux (entrée) dans un autre (sortie). Si nous appelons les évènements des suites d'entrée - "les causes" - et ceux de la suite de sortie "les effets" -, l'application C a alors les propriétés suivantes :

1) chaque effet a au moins une cause et au plus une cause par suite d'entrée ;
2) l'application conserve l'ordre : l'ordre des effets est induit de celui des causes ;
3) la durée des effets est uniquement fonction de la durée de leurs causes.

Dans cette partie, nous établissons les relations temporelles qui lient un "effet" à sa ou ses "causes", en fonction des incertitudes associées à la durée des évènements. L'étude se scinde en deux parties :

1) la convergence, où un seul effet peut avoir une ou plusieurs causes ; nous retrouverons les fonctions d'union, d'intersection et de mutuelle exclusion ;
2) la divergence, où plusieurs effets ont la même cause.

2.1. La convergence

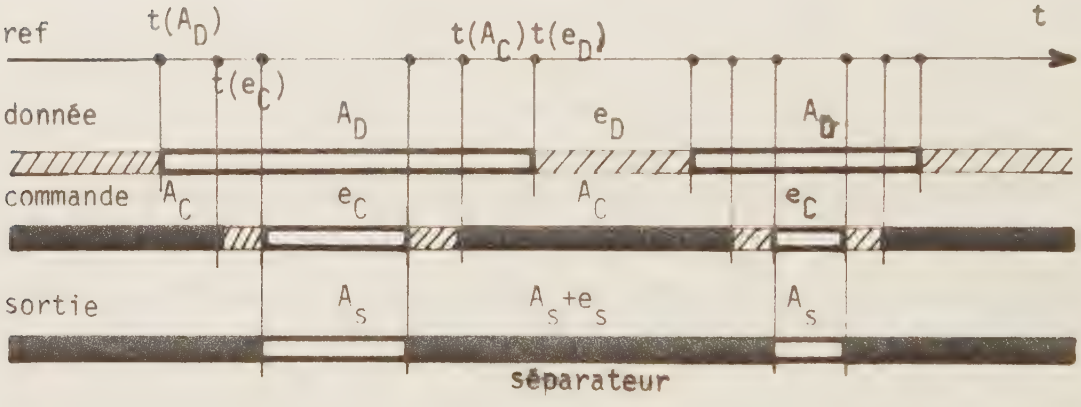
2.1.1. LE CONTROLE

Les composantes "indéfinies" d'un repère partiel peuvent avoir n'importe valeur. Relativement à l'alphabet associé, une composante indéfinie peut être composée de n'importe quel caractère ou de n'importe quelle suite de caractères. Or il importe parfois, en particulier pour la mémorisation, que le nombre de composantes définies d'un signal, relativement à un repère partiel, soit égal au nombre réel des composantes caractéristiques de ce signal, observé par rapport à un repère complet. Celà implique qu'une composante indéfinie (i) ne peut être que la "continuation" des composantes définies (a et b) qui l'enserrent. Autrement dit, soit i

telle que $a \frown i \frown b$ où a et b sont des composantes définies, alors 1) soit $i = a' \frown b'$, 2) soit $i = a' \frown e \frown b'$ où a'et a (resp. b' et b) appartiennent à la même partie de l'alphabet (ie : partition de l'ensemble des composantes) et où e est l'élément neutre (séparateur), par rapport à l'application envisagée.

Une application telle que la mémorisation nécessite donc la transformation d'un repère partiel quelconque (R_D) en un repère de ce type (R_S). Cette transformation peut être effectuée à l'aide d'une fonction extrêmale (ie. combinatoire). Il suffit d'effectuer l'union du repère R_D et d'un repère de commande (Rc) qui "injecte" un séparateur entre les composantes définies du repère R_D. Ces deux repères doivent alors vérifier entre eux les relations temporelles suivantes.

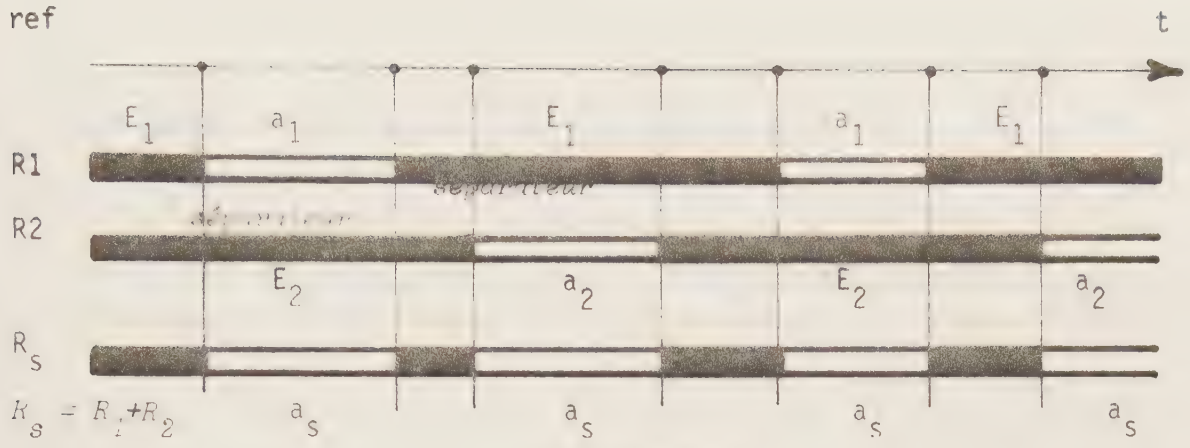
$$t(A_D) \leqslant t(e_C)$$
$$t(e_D) \geqslant t(A_C)$$



INJECTION DE SEPARATEURS

2.1.2. LE MULTIPLEXAGE

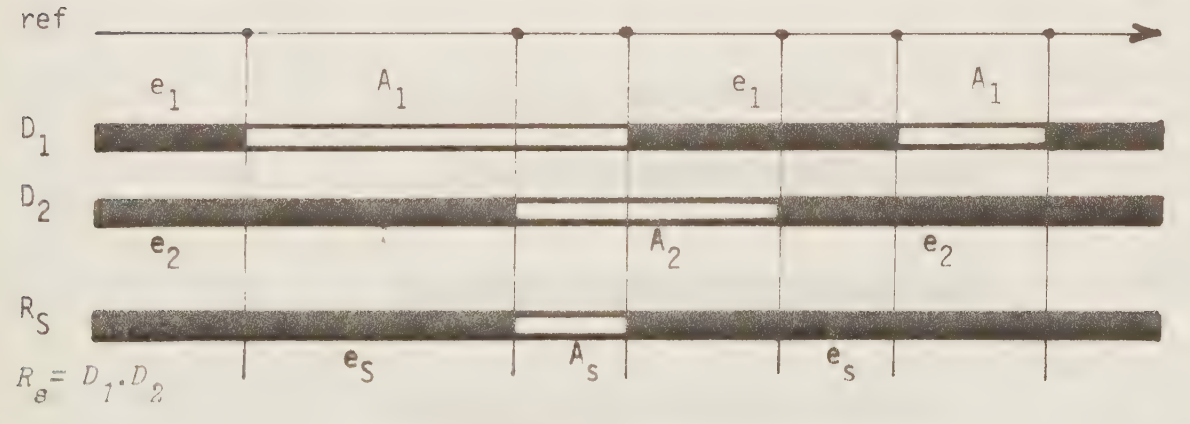
Le multiplexage peut être considéré comme le moyen de transmettre deux suites S_1 et S_2 d'"évènements" indépendants à l'aide d'un même support, c'est à dire d'un même signal S. Le multiplexage peut être réalisé à l'aide d'une fonction combinatoire, pourvu que les deux suites S_1 et S_2 soient mutuellement exclusives dans le temps. A chacune des suites S_1 et S_2 est associé un repère partiel, qui délimite de façon certaine les séparateurs, relativement à un référentiel indépendant. Les séparateurs correspondent aux composantes minimales (ie : neutre) de la fonction combinatoire extrêmale.



MULTIPLEXAGE

2.1.3 LA COMBINAISON

Cette application correspond à l'utilisation première des fonctions combinatoires : elle nécessite la simultanéité des composantes d'entrée. Si à chaque signal d'entrée est associé un repère partiel, relativement à un référentiel indépendant, la combinaison des signaux d'entrée n'est définie que sur l'intersection de ces repères.

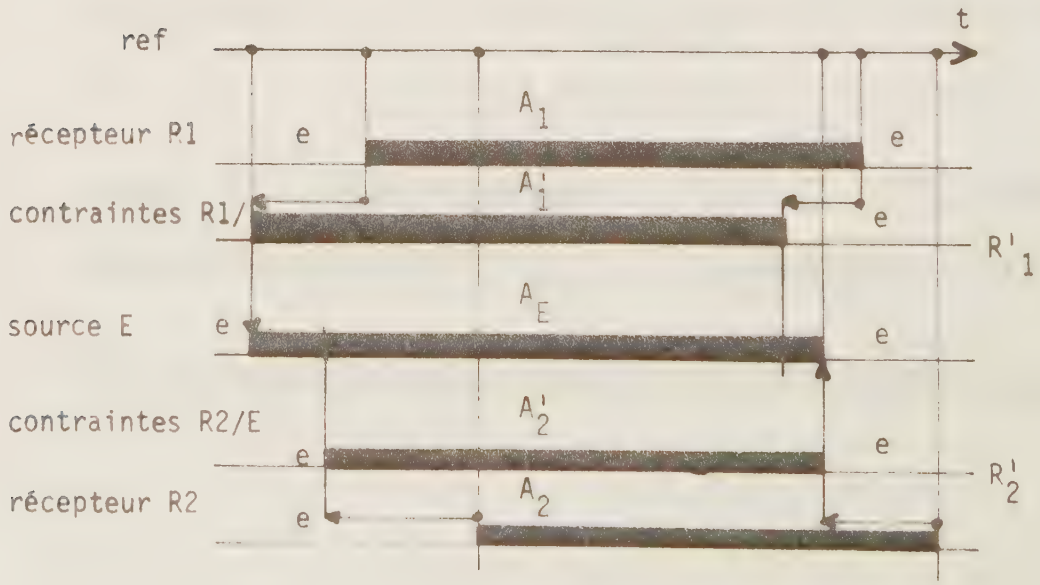


COMBINAISON

2.2. La divergence

Il arrive souvent que dans une machine, un même signal source soit transmis à plusieurs récepteurs. Lorsque les réceptions ne sont pas simultanées et n'ont pas la même durée, ou bien lorsque les retards de transmission ne sont pas identiques, le signal source doit vérifier certaines contraintes, afin que toutes les réceptions puissent être effectuées. Ces contraintes sont évaluées relativement à un référentiel indépendant. Chaque réception, à laquelle est associé un repère partiel, impose sur ce référentiel la date et la durée des composantes du signal source. Les caractéristiques globales de ce signal sont obtenues en effectuant l'union des composantes imposées pour chaque réception. Les dates de début et de fin des composantes définies du signal source correspondent alors au minimum et au maximum des dates imposées pour chaque réception.

Cette évaluation s'applique de la même façon à un éclatement pour lequel un signal source emprunte des chemins qui diffèrent par leur temps de retard de transmission pour se recombiner ensuite en un seul signal.



DIVERGENCE : $E = R'_1 + R'_2$

BIBLIOGRAPHIE

B I B L I O G R A P H I E

[ANC] F. Anceau
" Contribution à l'étude des systèmes hiérarchisés de ressources dans l'architecture des machines informatiques "
Thèse d'Etat - Université de Grenoble - Décembre 1974

[ARM] D.B. Armstrong
" A deductive method for simulating faults in logic circuits "
IEEE Trans. Comput. , vol. C 21 , pp 464 - 471 - Mai 1972

[ARS] J. Arsac
" Les langages sans étiquettes "
IP , n° 73.13 - Université Paris VI - 1973
" La Science Informatique "
Dunod - Paris - 1970

[BAR] M.R. Barbacci
" A comparison of Register Transfer Languages for describing computers and digital systems "
IEEE Trans. Comput. , vol. C 24 , pp 137 - 149 - Février 1975

[BAR] A. Barna
" High speed Pulse circuits "
Wiley Interscience - John Wiley & Sons, Inc - 1970

[BOO] T.L. Booth
" Sequential Machines and Automata Theory "
John Wiley & Sons, Inc - 1967

[BOU] N. Bourbaki
" Théorie des ensembles "
A.S.I. 1141 - Hermann - Paris - 1966

[BRI] L. Brillouin
" Science and Information Theory "
Academic Press - New York - 1956

[CAI] R.K. Caïn
" Automata theory, machines and languages "
Mc Graw Hill book company - Computer Science series - 1972

[CAR] M. Carvallo
" Monographie des treillis et algèbre de Boole "
Coll. Maths. Eco. - Gauthier Villars - Paris - 1962

[CAZ] M. Cazin
" Stabilité "
Encyclopédie Universalis - vol. 15 , pp 315 - 318 - 1968

[CHA] H.Y. Chang & S.G. Chappel
" Deductive techniques for simulating logic circuits "
Computer Journal - pp 52 - 59 - Mars 1975

[CHO] N. Chomsky & G.A. Miller
" L'analyse formelle des langues naturelles "
Gauthier Villars - Paris - 1968

[CHU] Y. Chu
" Digital computer design fundamentals "
Mc Graw Hill book company , Inc - 1962

[COS] O. Costa de Beauregard
" Le second principe de la science du temps "
Editions du Seuil - Paris - 1963

[CUR] H.A. Curtis
" A new type of double rank sequential machine "
IEEE Trans. Comput. , vol. C 22 , pp 796 - 803 - Aout 1973

[DEN] J.B. Dennis & S.S. Patil
" The description and realization of digital systems "
Colloque sur la conception et la maintenace des automatismes logiques
Toulouse - Septembre 1972

[DIE] D.L. Dietmeyer
" Logic design of digital systems "
Allyn and Bacon, Inc - Boston - 1971

[DRO] D. Dromard
" Représentation d'une procédure de transmission par des automates d'états finis "
Thèse de Docteur-Ingénieur - Université Paris VI - Juin 1974

[ELM] M.I. Elmasry & P.M. Thompson
" Two level emitter - function logic structures for logic-in - memory computers "
IEEE Trans. Comput. ,vol. C 24 , pp 250 - 257 - Mars 1975

[ETI] D. Etiemble
" Contribution à l'étude et à la modélisation des circuits logiques - Application à la logique de type TTL bas niveau "
Thèse de Docteur-Ingénieur - Université Paris VI - Juin 1974

[EVE] S. Even & A.R. Meyer
" Sequential Boolean equations "
IEEE Trans. Comput. , vol. C 18 , pp 230 - 240 - Mars 1969

[FRI] T.D. Friedman & S.C. Yang
" Methods used in an automatic logic design generator (ALERT) "
IEEE Trans. Comput. , vol. C 18 , pp 593 - 613 - Juillet 1969

[GER] G.B. Gerace & M. Vanneschi
" On the processing speed of microprogrammed systems "
Euromicro - preprints - Meeting June 1974 - Paris - Juin 1974

[HAR] F.H. Hardie & R.J. Suhocki
" Design and use of fault simulation for Saturn computer design "
IEEE Trans. Comput. , vol. EC 16 , pp 412 - 429 - Aout 1967

[HEB] J. Hebenstreit
" Générateur de microprogrammes "
Thèse d'Etat - Université Paris VI - Mai 1969
" Informatique théorique : circuits de logique séquentielle "
Cours C3 - IP , Université Paris VI - 1973

[HUF] D.A. Huffman
" The synthesis of sequential switching circuits "
Sequential machines - Addison Wesley Publishing Co - 1964

[HYD] S. Hyder
" A system for generating URDU/FARSI/ARABIC script "
IFIP 71 Congress Ljubljana - North Holland Publishing Co - 1972

[IBM] G.M. Amdahl et al.
" Improvements in or relating to electronic data processing machines "
IBM patent specification 1, 108, 800 - London - 1965

[LAD] J. Ladrière
" Système (épistémologie) "
Encyclopédie Universalis - vol. 15 , pp 685 - 687 - 1968

[LEV] J.M. Levy - Leblond
" Mécanique quantique "
Encyclopédie Universalis - vol. 13 , pp 860 - 868 - 1968

[MAV] J. Mavor
" Conception et technologie des circuits intégrés à transistors MOS "
Editions Eyrolles - Paris - 1975

[MAX] J. Max
" Méthodes et techniques de traitement du signal et applications aux mesures physiques "
Editions Masson & Cie - Paris - 1972

[MEA] G.H. Mealy
" A method for synthesizing sequential circuits "
Bell System Tech. Jour. , vol. 34 , pp 1045 - 1079 - Septembre 1955

[MIL] J. Millman & H. Taub
" Pulse, digital and switching waveforms "
International Stud. Ed. - Mc Graw Hill Book Co - 1965

[NOG] G.L.M. Noguez
" Présentation d'une technique : la microprogrammation "
Cours AEA - IP, Université Paris VI - 1974
" A quasi parallel hardware language emulated by a host high level language "
IEEE Conf. - COMPCON 75 fall - Washington - Septembre 1975
" Design of a microprogramming language "
MICRO 6 - ACM SIGMICRO - pp 145 - 155 - Septembre 1973
" A standardized microprogram sequencing control with a push down storage "
MICRO 5 - ACM SIGMICRO - pp 66 - 71 - Septembre 1972

[NOL] L. Nolin
" Formalisation des notions de machine et de programme "
Gauthier Villars - Paris - 1969
" Notions de formalisation "
Cours C3 - IP, Université Paris VI - 1968

[PAT] S.S. Patil
" Coordination of asynchronous events "
MAC TR - 72 - MIT - Boston - 1970

[PIL] R. Piloty
" Design verification at the register transfer language level "
Workshop on computer Hardware description languages - New Jersey - 1973

[RAV] Ravi Sethi & J.D. Ullman
" The generation of optimal code for arithmetic expressions "
Journal of ACM - vol. 17 , n° 4 - 1970

[SER] M. Servit
" Hazard correction in synchronous sequential circuits "
IEEE Trans. Comput. - vol. C 24 , pp 305 - 309 - Mars 1975

[SES] S. Seshu
" On an improved diagnosis program "
IEEE Trans. Comput. - vol. EC 14 , pp 76 - 79 - Février 1965

[SHA] C.E. Shannon
" A mathematical theory of Communications "
Bell System Tech. Jour. - Juillet et Octobre 1948

[SIF] J. Sifakis
" Modèles temporels des systèmes logiques "
Thèse de Docteur-Ingénieur - Université de Grenoble - Mars 1974

[SIM] J.C. Simon
" Traitement du signal : la représentation continue "
Cours C2 - IP, Université Paris VI - 1968

[SNA] J.P. Gray & C.R. Blair
" IBM's Systems Network Architecture "
Datamation - pp 51 - 56 - Avril 1975

[SPI] M.J. Spier
" Process communication prerequisite or the IPC-set up revisited "
Sagamore computer conference on parallel processing - 1973

[SUC] J. Suchard
" Etude physique des éléments logiques "
Cours C3 - IP, Université Paris VI - 1969/1975

[SZY] S.A. Szygenda, ed. & E.W. Thompson
" Digital systems simulation "
Computer Journal - pp 23 - 49 - Mars 1975

[UNG] S.H. Unger
" Asynchronous sequential circuits "
J. Wiley and Sons - New York - 1969

TABLE DES MATIERES

TABLE DES MATIERES

INTRODUCTION 1

CHAPITRE I : MOTIVATIONS 3

Présentation des systèmes séquentiels informatiques 5
Traitement séquentiel de l'information 5
Automate abstrait d'états 6
Domaine d'application usuel de la théorie des machines séquentielles 7

Problèmes temporels posés par les systèmes séquentiels 8
Etablissement et maintien d'un état 8
Repérage des états 10
"Déterminisme" des systèmes séquentiels 13
Influence de l'hypothèse de repérage unique 14

Les modèles existants des systèmes séquentiels 14
Modèles de simulation 15
Théorie des machines asynchrones 15
Théorie des machines synchrones 15
Les logiques booléennes temporelles 16

Le modèle proposé 17
Logique d'état 17
Logique temporelle 18
Les signaux comme éléments de base 19
Schéma d'analyse 19

Domaine d'application du modèle 21

CHAPITRE II : LE MODELE PROPOSE 25

La concaténation 27
Définitions préliminaires 28
Loi de concaténation 31
Signaux continus 33
Origine d'un signal 34

Les repères 37
Définition 37
Echelles de mesure 39

Algèbre des repères 43
Relation d'ordre 43
Lois de composition 45
Treillis distributif 46
La mémorisation 47

Les fonctions extrêmales 49
Définition 49
Porte logique 50
Initialisation logique 51
Jeu de fonctions extrêmales 52

Les isochrones 55
Définition 55
Relation de transfert 57
Etats 58
Synchronisme 59
Système séquentiel 60

note 1 : Alphabet Morse 62

note 2 : Algèbre de Boole et logique des événements 63

CHAPITRE III : APPLICATION DU MODELE AUX SYSTEMES A MEMOIRE:
ETUDE DES CELLULES DE MEMOIRE STATIQUES 65

LA BASCULE RS 69

La fonction idéale de mémorisation 69

Equation de mémorisation 71

Isochrone de mémorisation 73
Initialisation 73
Mémorisation 73
Durée 74
Repère intermédiaire I 75
Coordonnées 75

Automate séquentiel 77
Symétrie 77
Graphe des états 78

Cas limite de fonctionnement 80

Repérage 82

Régime oscillatoire 84

ETUDE DES BASCULES 86

Principe de l'étude 86
Contraintes temporelles 86
Mémorisation 87
Succession des états 88
Exemple d'analyse d'un circuit séquentiel 89

ECHANTILLONNAGE 93

	Signal de commande	93
	Caractéristiques d'un système d'échantillonnage	94
	Séparateurs	94
	Référence de temps	94
	Classification	95
	Echantillonnage permanent	95
	Complémentation en amont	97
	Complémentation en aval	98
	Contraintes temporelles	100
	Echantillonnage temporaire	101
	Echantillonnage sur "front avant"	101
	Echantillonnage sur "front arrière"	103
	Echantillonnage mémoriel	105
	TRANSPARENCE	110
	Les cellules à "double pont"	111
	Les bascules " maître-esclave"	113
note 1 :	La boucle de retour	115
note 2 :	Le tore magnétique	120
note 3 :	Résumé des caractéristiques des bascules statiques	124

CHAPITRE IV : APPLICATION DU MODELE AUX SYSTEMES A TEMPS PROPRE : ETUDE DES HORLOGES 129

Cycle de commandes 131
Notion de transfert élémentaire 131
Synthèse d'un cycle de commandes 133
Exemple 138

Diagramme des temps 145
Repérage et mesure des caractéristiques temporelles 145
Domaine temporel arborescent d'une commande 147
Isochrone de contraintes 150
Concaténation des domaines temporels 158
Superposition de séquences de commande 160

Domaines temporels 161
Domaines synchrones et asynchrones 161
Couplage 162
Découplage temporel 165

Structure d'une horloge 169
Horloges monophasées 169
Horloges multiphasées 169
Exemple 170

note 1 : Les circuits oscillateurs 175

CONCLUSION 179

Analyse des systèmes séquentiels 181

Synthèse des systèmes séquentiels 184

Limites du modèle 185

ANNEXE : APPLICATION DU MODELE AUX SYSTEMES COMBINATOIRES 187

Préambule 188
Modèle synthétique des éléments logiques 188
Les paramètres temporels 189

Le retard simple 193
Caractéristiques physiques 193
Repérage 194
Composantes parasites 195
But recherché 196
Relations temporelles 197
Cumul des temps de retard 201
Distance invariante 201

Les fonctions combinatoires 202
La convergence 203
La divergence 206

BIBLIOGRAPHIE 207